Working Report 2013-01

M>OC> User's Manual: Migration Analysis of Radionuclides in the Far Field

Scott Painter[1]
Computational Earth Sciences Group
Earth and Environmental Sciences Division
Los Alamos National Laboratory

James Mancillas
Center for Nuclear Waste Regulatory Analyses
Southwest Research Institute®

December 2013

[1]Previously at Center for Nuclear Waste Regulatory Analyses, Southwest Research Institute.

Working Reports contain information on work in progress
or pending completion.

ABSTRACT

The computer code Migration Analysis of Radionuclides in the Far Field (MARFA) uses a particle-based Monte Carlo method to simulate the transport of radionuclides in a sparsely fractured geological medium. The algorithm uses non-interacting particles to represent packets of radionuclide mass. These particles are moved through the system according to rules that mimic the underlying physical transport and retention processes. The physical processes represented in MARFA include advection, longitudinal dispersion, Fickian diffusion into an infinite or finite rock matrix, equilibrium sorption, decay, and in-growth. Because the algorithm uses non-interacting particles, the transport and retention processes are limited to those that depend linearly on radionuclide concentration. Multiple non-branching decay chains of arbitrary length are supported, as is full heterogeneity in the transport and retention properties. Two variants of the code are provided. These two versions differ in how particles are routed through the computational domain. In MARFA 3.2.3, transport is assumed to occur along a set of trajectories or pathways that originate at radionuclide source locations. The trajectories are intended to represent the movement of hypothetical, advectively transported groundwater tracers and are typically calculated by pathline tracing in a discrete fracture network flow code. The groundwater speed and retention properties along each pathway may change in time, but the pathway trajectories are fixed. MARFA 3.3.1 allows the transport effects of changing flow directions to be represented by abandoning the fixed pathways and performing node routing within MARFA.

Keywords: Monte Carlo method, Transport modeling, Diffusion, Matrix diffusion, Advection, Dispersion, Groundwater flow, Transport of radionuclides, Pathlines, Trajectories, Retention, Transport resistance.

MARFA-kulkeutumismallinnusohjelman (*Migration of Radionuclides in the Far Field*) käyttöopas

TIIVISTELMÄ

Tietokoneohjelma *Migration of Radionuclides in the Far Field* (MARFA) mallintaa radionuklidien kulkeutumista harvaan rikkonaisessa kiteisessä kallioperässä. Tässä se käyttää Monte Carlo -menetelmää. Algoritmissa keskenään vuorovaikuttamattomat partikkelit edustavat radionuklidien massaa. Nämä partikkelit liikkuvat systeemin läpi fysikaalisen kulkeutumisen ja pidättymisprosessien mukaisesti. MARFAn kulkeutumismallinnukseen sisältyvät fysikaaliset prosessit ovat advektio, pitkittäinen dispersio, sekä rajoitettu että rajoittamattoman (Fickin) matriisidiffuusio, tasapainosorptio, ja radioaktiivinen hajoaminen ja sisäänkasvu. Koska algoritmi perustuu oletukseen, että partikkelit eivät vuorovaikuta keskenään, kulkeutumis- ja pidättymisprosessit rajoittuvat niihin, jotka riippuvat radionuklidien pitoisuudesta lineaarisesti. Tietokoneohjelma tukee sekä useita haarautumattomia hajoamisketjuja että heterogeenisiä kulkeutumis- ja pidättymisominaisuuksia.

Tietokoneohjelmasta on tuotettu kaksi versiota. Ne poikkeavat toisistaan siinä miten partikkelit reitittyvät mallinnusalueen läpi. MARFAn versiossa 3.2.3 kulkeutumisen oletetaan tapahtuvan pitkin radionuklidien lähdepisteistä alkavien virtausreittien joukkoa. Näiden reittien tarkoitus on edustaa hypoteettisia, pohjaveden advektion mukana liikkuvia merkkiaineita ja tyypillisesti ne kuvataan hydrogeologisella rakoverkkomallilla laskettujen virtaviivojen avulla. Tässä MARFAn versiossa virtausnopeus ja pidättymisominaisuudet voivat muuttua ajan myötä, mutta virtaviivat sinänsä pysyvät muuttumattomina. Sen sijaan MARFAn versiossa 3.3.1 tätä rajoitusta ei ole vaan virtaviivat voivat muuttua ajan myötä ja partikkeleiden reititys tehdään MARFAn sisällä.

Avainsanat: Monte Carlo -menetelmä, kulkeutumismallinnus, diffuusio, matriisidiffuusio, advektio, dispersio, pohjaveden virtaus, radionuklidien kulkeutuminen, virtaviivat, virtausreitit, pidättyminen, kulkeutumisvastus.

TABLE OF CONTENTS

ABSTRACT
TIIVISTELMÄ

FIGURES 3
FOREWORD 5
1 INTRODUCTION 7
1.1 MARFA Versions 8
1.2 Geosphere Transport Analysis Workflow With MARFA 8
2 MARFA 3.2.3 ALGORITHMS 13
2.1 Particle Tracking Algorithm for a Single Species and Single Segment 13
2.2 Pathways with Multiple Segments 16
2.3 Multiple Pathways 16
2.4 Decay Chains 17
2.5 Pathways with Time-Dependent Velocity or Retention Parameters 18
2.5.1 Probability Density for Depth in the Matrix 19
2.5.2 Probability Density for Time of Return 20
2.6 Downscaling Algorithm 21
2.7 Reconstruction of the Breakthrough Curves 22
3 MARFA 3.3.1 ALGORITHMS 25
3.1 Node Routing 25
3.2 Stuck Particles 26
3.3 Stranded Particles 26
3.4 3Node Renumbering 26
4 INSTALLING AND RUNNING MARFA 27
5 VERIFICATION TESTS 29
5.1 Unlimited Diffusion 29
5.2 Limited Diffusion 30
5.3 Multiple Pathways 30
5.4 Low Flow Period with Unlimited Diffusion 31
5.5 Flow Period with Limited Diffusion 32
5.6 High Flow Period with Unlimited Diffusion 34
5.7 Changes in Flow and Sorption Properties 34
5.8 Limited Matrix Diffusion in Tabular Form 34
5.9 Diffusion into a Three-layer Matrix 35
5.10 Particle Splitting 35
6 VERIFICATION TESTS FOR VERSION 3.3.1 41
6.1 Tests of the Retention Model 41

6.2 Flow Change Test 42
6.3 Multiple Sources and Multiple Pathways Test 43
7 EXAMPLE SIMULATIONS 45
8 SUMMARY 53
REFERENCES 55
APPENDIX A: MARFA 3.2.3 INPUT 57
APPENDIX B: MARFA 3.3.1 INPUT 73
APPENDIX C: MARFA OUTPUT 77

FIGURES

1. Geosphere transport analysis workflow based on MARFA and CONNECTFLOW 10

2. Results of Verification Test 1 30

3. Results of Verification Test 2 31

4. Results of Verification Test 3 31

5. Results of Verification Test 4 32

6. Results of Verification Test 5 33

7. Results of Verification Test 6 34

8. Results of Verification Test 7 36

9. CONNECTFLOW trajectories forming the pathways for the MARFA example simulation 37

10. Valid pathways used in the example simulation color coded by cumulative radionuclide discharge as calculated by MARFA 38

11. Radionuclide discharge for ^{237}Np for example simulations with and without flow field changes due to glacier passage 39

12. Result of Version 3.3.1 Verification Test 1 42

13. Result of Version 3.3.1 Verification Test 2 43

14. Result of Version 3.3.1 Verification Test 3 44

15. CONNECTFLOW trajectories forming the pathways for the MARFA example simulation 46

16. Valid pathways used in the example simulation color coded by cumulative radionuclide discharge as calculated by MARFA 46

17. Radionuclide discharge for ^{237}Np for example simulations with and without flow field changes due to glacier passage 47

18. Radionuclide discharge for an example demonstrating the use of particle splitting passage 48

19. Fracture Network for the Forsmark 100-m Block Simulation. 50

20. Transport Pathways as calculated by CONNECTFLOW for the Forsmark 100-m Block Simulation 50

21. Result of the Forsmark 100-m Block Simulation. 51

FOREWORD

Initial versions of the Migration Analysis of Radionuclides in the Far Field (MARFA) computer code were developed by the Center for Nuclear Waste Regulatory Analyses (CNWRA®) for the Swedish Nuclear Fuel and Waste Management Company and Posiva Oy under Contract Number SWRI–2044832.

MARFA improvements appearing in Versions 3.2.3 and 3.3.1 were authored by Los Alamos National Laboratory (LA-CC-11-089). The auxiliary application Three Layer Diffusion, Version 1, which calculates certain input files to MARFA was also authored by Los Alamos National Laboratory (LA-CC-11-088). Development of the Improvements to the MARFA Code and Three Layer Diffusion were funded by the Swedish Nuclear Fuel and Waste Management Company and Posiva Oy under a Non-Federal Entities Work-for-Others Agreement (NFE-WFO)) No. FIA-10-024.

This document has been approved for general release (LA-UR-12-27042).

1 INTRODUCTION

The computer code Migration Analysis of Radionuclides in the Far Field (MARFA) uses a particle-based Monte Carlo method to simulate the transport of radionuclides in a sparsely fractured geological medium. Transport in sparsely fractured rock is of interest because this medium may serve as a barrier to migration of radionuclides to the accessible environment. The physical processes represented in MARFA include advection, longitudinal dispersion, Fickian diffusion into an infinite or finite rock matrix, equilibrium sorption, decay, and in-growth. Multiple non-branching decay chains of arbitrary length are supported.

MARFA uses the particle on random streamline segment algorithm [Painter et al. 2006], a Monte Carlo algorithm combining time-domain random walk methods [Painter et al. 2008] with pathway stochastic simulation [Painter and Cvetkovic 2005]. The algorithm uses non-interacting particles to represent packets of radionuclide mass. These particles are moved through the system according to rules that mimic the underlying physical transport and retention processes. The set of times required for particles to pass through the geological barrier are then used to reconstruct discharge rates (mass or activity basis). Because the algorithm uses non-interacting particles, the transport and retention processes are limited to those that depend linearly on radionuclide concentration. Nonlinear processes such as solubility-limited transport or aqueous speciation are not represented.

The MARFA code is specifically designed to work with output from discrete fracture network (DFN), continuous porous medium (CPM), or nested DFN/CPM flow models. MARFA has capabilities to stochastically simulate relevant transport properties along the pathways. This mode of operation is useful when the pathways have been determined from an upscaled CPM flow calculation. Upscaled CPM flow models are generally regarded as adequate for representing regional-scale groundwater flow. Transport under field conditions is, however, much more sensitive than flow to local variations in velocity, and it is not clear that global transport can be adequately represented with an upscaled CPM representation of the velocity fields. The pathway stochastic simulation option helps to recover, in a statistical sense, the subgrid velocity variations that are lost in an upscaled CPM flow representation. Thus, the pathway simulation option represents an example of statistical downscaling. The stochastic pathway simulation (downscaling) is based on the algorithm described by Painter and Cvetkovic [2005]. The downscaling algorithm is itself a type of random walk that incorporates persistence to account for sequential correlation in properties along the pathway.

The Monte Carlo algorithm in MARFA produces output in the form of particle arrival times at pathway end points. Cumulative mass discharge (cumulative breakthrough) at a given time can be readily constructed from the arrival times by simply identifying the amount of mass arriving before the specified time. This procedure is equivalent to estimating cumulative probability distributions from a set of random samples. Estimating the mass discharge rate (instantaneous breakthrough) is analogous to the more difficult task of estimating probability density from a set of random samples. MARFA uses a two-step post-processing procedure to reconstruct the instantaneous breakthrough curves. The algorithm used in the post processor is an adaptive kernel

estimation method with kernel width calculated from an initial or "pilot" estimate obtained from a nearest neighbor method. This adaptive two-step method was chosen because of its capability to provide stable and smooth density estimations for a wide variety of data distributions, while maintaining sensitivity to local details. The algorithm is described in detail by Silverman [1986].

1.1 MARFA Versions

Two variants of the code are provided. These two versions differ in how particles are routed through the computational domain.

In the MARFA 3.2 series, transport is assumed to occur along a set of trajectories or pathways that originate at radionuclide source locations. The trajectories are intended to represent the movement of hypothetical, advectively transported groundwater tracers and are typically calculated by pathline tracing in a DFN or DFN/CPM flow code. The groundwater speed and retention properties along each pathway may change in time, but the pathway trajectories are fixed. An option is provided to take the input from CONNECTFLOW verbose pathline (PTV) files, which are generated within CONNECTFLOW using conventional streamline tracing for continuum porous medium (CPM) regions and node network routing for the discrete fracture network (DFN) regions [Hartley and Holton 2003].

Given the long time frames considered in assessments of potential high-level nuclear waste repositories in Sweden and Finland, significant changes in groundwater flow direction are expected due to glacial rebound and changing climate states in the future. For example, land rise due to glacial rebound and the passage of future glaciers may generate significant changes in groundwater discharge locations. Such scenarios cannot be fully evaluated with fixed pathways. MARFA 3.3.1 allows the transport effects of changing flow directions to be represented by abandoning the fixed pathways and performing node routing within MARFA. With this modification to the MARFA algorithms, the CONNECTFLOW pathway PTV file is no longer needed; instead, the CONNECTFLOW node-routing (PTH) file is required. Version 3.3.1 is intended to be an alternative to Version 3.2.3, not a replacement. That is, both versions are actively maintained and in use. Both versions are described in this document.

1.2 Geosphere Transport Analysis Workflow with MARFA

The MARFA software is intended to be used in combination with the CONNECTFLOW [Hartley and Holton 2003] software or similar DFN simulator as part of an integrated workflow. An example workflow for geosphere transport analysis using CONNECTFLOW and MARFA 3.2.3 is represented graphically in Figure 1. This workflow is similar to those used previously [Hartley et al. 2004], except that the MARFA software is used in combination with CONNECTFLOW discrete-fracture network (DFN) particle tracking simulations to recover the effects of subgrid scale velocity variations. Although subgrid-scale velocity variations can be important in determining global transport characteristics of a geosphere barrier, they are lost in the CONNECTFLOW continuous porous medium (CPM) representation of the regional scale groundwater system.

In the workflow shown in Figure 1, two types of trajectory data are passed from CONNECTFLOW to MARFA: regional-scale trajectory data from the nested DFN/CPM model, and generic subgrid trajectories. Both types of trajectory data are generated by advection-only particle tracking. Data are required for each segment of each trajectory. Here "segment" refers to that part of a trajectory that passes through a single fracture in the DFN regions. In the CPM region, the granularity of the trajectory is somewhat arbitrary as long as the segments are small enough to resolve the trajectory.

The trajectories from the regional-scale nested DFN/CPM model connect a set of source locations to a monitoring boundary, as in the previous transport workflow [Hartley et al., 2004]. A typical modeling case may include trajectories that pass through explicit DFN representations of hydraulic rock domains; repository tunnels and near-surface soils represented as CPM regions; deformation zones represented as deterministic features; and hydraulic rock domains represented as CPMs or Equivalent CPMs (ECPMs). In the DFN regions, repository tunnels, and near-surface soils, the trajectory/pathway would be considered deterministic with properties specified by CONNECTFLOW. When a hydraulic rock domain is represented as an ECPM, MARFA is also capable of running in an optional stochastic mode, wherein trajectory properties are generated stochastically. The stochastic mode was explicitly designed to help capture transport effects of subgrid velocity variability that would have otherwise been lost when hydraulic rock domains are treated as ECPM.

To run MARFA, one or more hydrogeological zones (rock types) must also be defined. A rock type has an associated radionuclide retention model, a set of retention parameters, and a flag indicating whether it is to be treated deterministically or stochastically. Flow-related fracture properties are implicit in the trajectory calculation and are not part of the rock type input. For a typical application, a rock type would be defined for an explicit DFN region, the repository tunnels, each of the hydrogeological units represented as an ECPM, each near-surface soil layer, and the large-scale deterministic features.

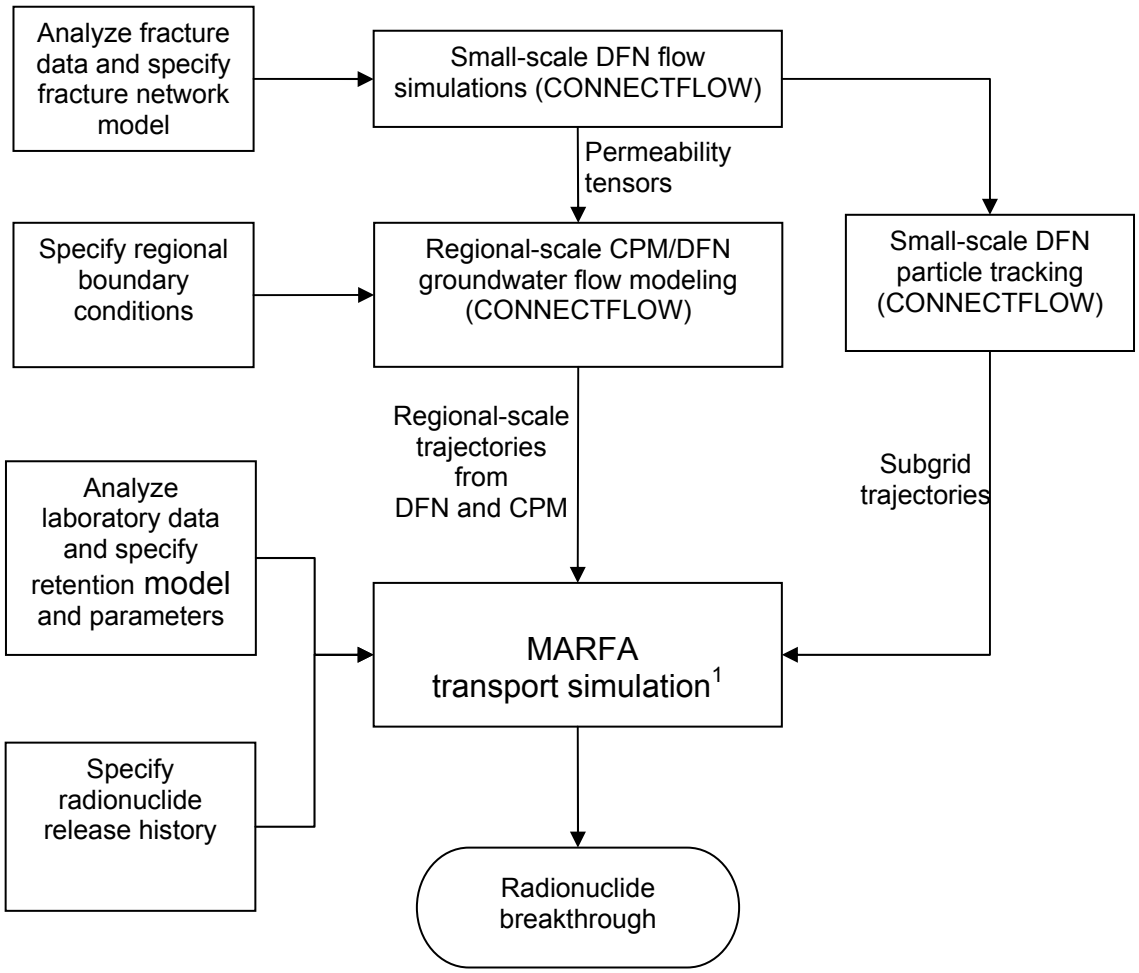
Analyze fracture data and specify fracture network model
Small-scale DFN flow simulations (CONNECTFLOW)
Permeability tensors
Specify regional boundary conditions
Regional-scale CPM/DFN groundwater flow modeling (CONNECTFLOW)
Small-scale DFN particle tracking (CONNECTFLOW)
Regional-scale trajectories from DFN and CPM
Subgrid trajectories
Analyze laboratory data and specify retention model and parameters
MARFA transport simulation[1]
Specify radionuclide release history
Radionuclide breakthrough

Figure 1. *Geosphere transport analysis workflow based on MARFA and CONNECTFLOW. This workflow presumes steady-state groundwater velocities, which allows a one-way flow of information from CONNECTFLOW to MARFA.*

[1]Each trajectory from the CONNECTFLOW nested CPM/DFN flow model may pass through a DFN region and a CPM region. Along the CPM part of the trajectory, MARFA has an option to operate in a stochastic mode that combines the trajectory information with subgrid trajectories to recover, in a statistical sense, the effects of subgrid variability. The subgrid trajectories are not used in the DFN region. For that part of the trajectory, MARFA operates in a deterministic pathway mode.

The two downscaling (streamline extrapolation) algorithms described by Painter et al. [2006] are used in the MARFA software to stochastically simulate subgrid-scale velocity variations when a stochastic rock type is specified. These two algorithms require a set of subgrid trajectories from CONNECTFLOW DFN simulations. These subgrid trajectories are generated by advection-only particle tracking on relatively small DFN simulations. The procedure is somewhat analogous to the numerical upscaling procedure used to establish permeability tensors for ECPM models [Hartley et al. 2004] in that information is extracted from small-scale DFN simulations and used later to

improve the representation of transport processes in a continuum model. The spatial scale required for the DFN simulations used to produce the subgrid trajectories is highly dependent on the properties of the DFN model. Selection of the spatial scale must balance several considerations. Ideally, the spatial scale would be similar in magnitude to the largest stochastically simulated fracture, much smaller than the modeled region, and the same as the CONNECTFLOW ECPM cell size. Ultimately, sensitivity studies should be used to select the required scale of the DFN simulations used to construct the subgrid trajectories.

If a single isotropic DFN model is representative of the entire ECPM model region, then a single set of subgrid trajectories is sufficient for MARFA. The single set might be composed of a small number (~ 10) of DFN realizations and a large number (a few thousand) trajectories.

For anisotropic networks, the trajectory statistical properties will, in general, depend on the direction of the macroscopic gradient relative to the DFN principal direction. The strategy for handling this situation is to consider a small number of directions for the macroscopic gradient and produce a set of subgrid trajectories for each. That is, for anisotropic networks, several sets of subgrid trajectories are required, one for each direction. Similarly, if different DFN models apply in different hydrostratigraphic units, then the DFN simulations must be repeated for each DFN model. The total number of required subgrid trajectory sets is equal to the number of directions times the number of stochastic rock types.

2 MARFA 3.2.3 ALGORITHMS

MARFA 3.2.3 uses a time-domain particle tracking algorithm to simulate transport along a set of 1-D pathways. These calculations produce a set of arrival times at each pathway terminus. A post-processing step based on an adaptive kernel method is then used to reconstruct mass breakthrough curves from the arrival times. Details of the particle-tracking algorithms are given by Painter et al. [2006, 2008]. A summary is provided here. The algorithm is first presented for a single species and a single pathway segment that is characterized by steady groundwater velocity and retention properties. Generalizations to multiple segments, multiple trajectories, multiple species linked through a decay chain, and transient pathway properties are then discussed. Finally, the algorithm for reconstructing the breakthrough curve is summarized.

2.1 Particle Tracking Algorithm for a Single Species and Single Segment

We are interested in mass discharge at a pathway (section) or segment outlet versus time. The mass discharge rate [mols/T] can be written as

$$r_{out}(t) = \int_0^t f_{tran}(t-t')\, r_{in}(t')\, dt' \tag{1}$$

where r_{in} is the rate at which mass is introduced into the pathway [mols/T]. The input rate can also be written as $r_{in}(t) = S_0 f_{in}(t)$ where S_0 is the total source strength and $f_{in}(t)$ is the input rate normalized as a probability density. The kernel in the convolution, f_{tran}, is the transit or residence time distribution for packets of mass in the pathway. The kernel may also be interpreted as the discharge rate due to a Dirac-δ input.

A well-known result from chromatography [Villermaux, 1987] is to compute f_{tran} as

$$f_{tran}(t_{tran}) = \int_0^\infty f_{ret}(t_{tran}-\tau)\, f_\tau(\tau)\, d\tau \tag{2}$$

where $t_{tran} - \tau = t_{ret} \geq 0$ is the retention time, which has probability density f_{ret}. In chromatography, this expression is usually written in the Laplace domain. Note also that in Eq. (2) the upper limit of integration extends to infinity instead of t_{tran} because we include a Heaviside function centered at zero in our definition of the retention time distribution. The basic assumption in Eq. (2) is that the tracer concentration is relatively low (dilute systems) such that retention (mass transfer) processes are linear (in a generalized sense). In Eq. (2), τ is the water residence time, and f_τ is the water residence time density. The effect of longitudinal dispersion is accounted for by an appropriate choice of f_τ. An explicit form for $f_\tau(\tau)$ is given by Painter [2008].

Equation (2) neglects variability in retention. It has been shown previously [Dagan and Cvetkovic 1996; Cvetkovic et al. 1998, 1999; Cvetkovic and Haggerty 2002] that diffusive transfer between mobile and immobile states, and hence retention, depends on velocity. Thus, in heterogeneous media, retention is generally variable, similar to τ,

even when the chemical retention parameters are spatially constant. For a broad class of retention models, retention variability can be parameterized by a single velocity-dependent (random) parameter denoted β, and a set of uniform physicochemical retention parameters (factorized case, in Cvetkovic and Haggerty 2002). Thus, two velocity-dependent parameters characterize the transport: the nonretarded (water) residence time τ and the transport resistance parameter β. The transport resistance parameter[1] depends on the choice of retention model. For the case of matrix diffusion, β is the quantity $1/(b\,v)$ integrated along the pathway segment, where b is fracture half-aperture and v is fluid velocity. Regardless of the retention model, τ and β are highly correlated through a shared dependence on the velocity field. In fact, if variability in mobile porosity (or fracture aperture) is neglected, $\beta \propto \tau$.

We generalize Eq. (2) to properly account for the coupling between retention and longitudinal dispersion by including longitudinal dispersion for both τ and β

$$f_{tran}(t_{tran}) = \int_0^\infty \int_0^\infty f_{ret}(t_{tran} - \tau \mid \beta) f_{\beta|\tau}(\beta \mid \tau) f_\tau(\tau) d\tau d\beta \tag{3}$$

where $f_{\tau,\beta}(\tau,\beta) = f_{\beta|\tau}(\beta \mid \tau) f_\tau(\tau)$. In other words, τ and β become random variables with an appropriately specified joint density $f_{\tau,\beta}(\tau,\beta)$. This distribution depends on $\bar{\tau}$ and $\bar{\beta}$, which are values for τ and β in the absence of longitudinal dispersion and regarded as properties of the pathway section (segment), but this dependence is not specifically included in the $\bar{\tau}$ notation. Spatial variability within each segment is neglected in MARFA, thus implying $\frac{\tau}{\bar{\tau}} = \frac{\beta}{\bar{\beta}}$ and

$$f_{\beta|\tau}(\beta \mid \tau) = \delta\left(\beta - \tau\bar{\beta}/\bar{\tau}\right) \tag{4}$$

where δ is the Dirac-δ function.

The mass discharge rate or breakthrough can now be written as

$$r_{out}(t) = S_0 \int_0^\infty \int_0^\infty f_{ret}\left(t - t_{in} - \tau \mid \tau\bar{\beta}/\bar{\tau}\right) f_\tau(\tau)\; f_{in}(t_{in}) d\tau \;\; dt_{in} \tag{5}$$

and the cumulative tracer breakthrough curve as

$$R_{out}(t) = S_0 \int_0^\infty H(t - t_{ar}) \int_0^\infty \int_0^\infty f_{ret}\left(t_{ar} - t_{in} - \tau \mid \tau\bar{\beta}/\bar{\tau}\right)\; f_\tau(\tau)\; f_{in}(t_{in}) d\tau \;\; dt_{in}\, dt_{ar} \tag{6}$$

[1]The transport resistance parameter denoted β here is commonly denoted F within other SKB documents.

where $H(\text{-})$ is the Heaviside function. Recall that our definition of retention time includes a Heaviside function at $t_{\text{ret}} = 0$, which is why the upper limits of integration are infinity in Eqs. (5) and (6). A Monte Carlo estimate of Eq. (6) is

$$\hat{R}_{out}(t) = \frac{S_0}{N_{part}} \sum_i H\left(t - t_{ar,i}\right) \tag{7}$$

Here $t_{ar,i}$ is one of N_{part} samples from the arrival time distribution, which has density

$$f_{ar}\left(t_{ar}\right) = \int_0^\infty \int_0^\infty f_{ret}\left(t_{ar} - t_{in} - \tau \mid \tau\overline{\beta}/\overline{\tau}\right) f_\tau(\tau) f_{in}\left(t_{in}\right) d\tau\, dt_{in} \tag{8}$$

The time-domain particle tracking algorithm used in MARFA is formally equivalent to a Monte Carlo sampling of Eq. (6). In the following, $\overline{\tau}$ and $\overline{\beta}$ are known – these are properties of the segment. The algorithm for the cumulative discharge curve is now assembled as follows:

(1) Sample a random start time t_{in} from the normalized source $f_{in}\left(t_{in}\right)$.

(2) Sample a τ value based on longitudinal dispersion. The appropriate distribution has density $\overline{\tau}\, f_\tau(\tau) = \sqrt{\frac{\eta}{4\pi}}\left(\tau'\right)^{-3/2} \exp\left(-\frac{\eta}{4}\frac{\left(1-\tau'\right)^2}{\tau'}\right)$ where $\tau' = \frac{\tau}{\overline{\tau}}$, $\eta = \frac{\ell}{\alpha}$, ℓ is the length of the segment, and α is the dispersivity. An algorithm for sampling this is described in the appendix of Painter at al. [2008].

(3) Calculate a β value from $\beta = \tau\overline{\beta}/\overline{\tau}$. Note that this step properly accounts for the interaction between retention and longitudinal dispersion.

(4) Sample a retention time t_{ret} from $f_{ret}\left(t_{ret} \mid \beta\right)$. Retention time distributions for important retention models are compiled in Painter et al. [2008]. For computational efficiency, the sampling scheme in MARFA uses quantiles that are precomputed from analytical or semianalytical cumulative distributions and stored as lookup tables. Retention time distributions for diffusion and sorption in a homogeneous matrix of unlimited or limited extent are based on those in Table 1[2] of Painter et al. [2008]. Starting in Version 3.2.3, a helper application (see Appendix A-4) is provided that computes retention time distributions based on a 3-layer matrix model for use in MARFA. The 3-layer matrix model is based on Cvetkovic [2009]. To sample from the retention time distribution, a random number uniformly distributed between 0 and 1 is selected and the corresponding quantile is then interpolated from that lookup table.

[2]It is important to note that the diffusion coefficient D used by Painter et al. [2008] is the pore diffusion coefficient and is related to the effective diffusion coefficient D_{eff} by $D = D_{eff}/\theta$ where θ is matrix porosity.

(5) Calculate the particle arrival time as $t_{ar} = t_{in} + \tau + t_{ret}$. This value represents one sample from the arrival time distribution.

(6) For a given time t, if $t_{ar} < t$ the particle contributes an amount S_0 / N_{part} to the cumulative mass discharge.

(7) Repeat from Step 1 a total of N_{part} times.

In practice, the arrival time and associated mass for each particle may be recorded and used in a postprocessing step to construct an approximation to $R_{out}(t)$ and $r_{out}(t)$. Reconstruction of $R_{out}(t)$ is simply a matter of summing the mass that arrives before a given time, as in step 6 and Eq. (7). Reconstruction of $r_{out}(t)$ is analogous to reconstructing the probability density from a set of samples and is more difficult than the reconstruction of the $R_{out}(t)$. A kernel method for reconstruction of $r_{out}(t)$ is discussed in 2.7.

2.2 Pathways with Multiple Segments

The algorithm described in Section 3.1 readily generalizes to a pathway with multiple segments. Because the discharge from one segment is the input to the following segment in a 1-D pathway, the algorithm may simply be applied recursively to model transport on a pathway composed of multiple segments. When a particle exits a segment, it is restarted in the subsequent segment without resetting the particle clock. Therefore the arrival time at the pathway terminus includes the contribution from each segment on the pathway.

2.3 Multiple Pathways

MARFA supports an arbitrary number of sources. A source may be regarded as a single waste canister or as a group of canisters with identical release histories. Each source may be connected to the monitoring boundary by one or multiple pathways. In the case that multiple pathways originate from a single source, the pathways are presumed to be equally probable. Pathways are not necessarily independent because pathways originating from a single waste canister location may pass through many of the same fractures.

To accommodate multiple pathways, MARFA randomly picks a source–nuclide combination and a pathway each time a particle is released. The probability of sampling a given source–nuclide combination is proportional to the associated cumulative source strength in moles. Once the source is selected, MARFA selects randomly from the pathways associated with that location and uses that pathway until the particle reaches the pathway terminus. The sampling approach may be altered by specifying a relative importance parameter for each nuclide. Higher values for the importance factor cause a radionuclide to be sampled more frequently; a corresponding reduction in statistical weight is then used to avoid biasing the breakthrough estimates. The importance

parameter may be used, for example, to limit the number of particles assigned to radionuclides with high source rates but low dose conversion factors.

MARFA provides two options for determining the time at which a particle is released once a source–nuclide combination has been selected. The default method is to give each particle of a given source–nuclide combination equal statistical weight and make the rate of releasing particles proportional to the source strength history (mol/yr). An alternative to the equal-weight method is available starting in Version 3.2.2. In this method, particle release times are sampled uniformly in time. Each particle is then given a statistical weight proportional to the source strength at the sampled time of release. Both methods produce unbiased statistical estimates of the breakthrough curves. The uniform-weight method provides more efficient (in the sense of requiring fewer particles to achieve the same level of uncertainty) estimates of cumulative breakthrough. The uniform-in-time method produces more efficient estimates of the leading edge of a breakthrough curve when the source history has a large dynamic range. All results shown in this document use the default method.

The MARFA approach of associating a particle randomly with one of the pathways is appropriate because of the Monte Carlo nature of the algorithm. When the quantity of interest is breakthrough for an ensemble of pathways, MARFA's Monte Carlo approach has large computational advantages compared with a deterministic method. The advantage arises because it is not necessary to accurately resolve the breakthrough for each pathway in order to have an accurate representation of the ensemble breakthrough.

2.4 Decay Chains

The complicating factor in transport analysis of radionuclide chains is that sorption and other physicochemical retention parameters are, in general, different for different members of the chain, which causes the retention-time distributions to be different. Initial tests of the time-domain algorithms revealed that significant biases may be introduced if decay and in-growth are not handled carefully. For example, the simplest algorithm, ignoring the transformation event until the particle completes the segment, may overestimate or underestimate the breakthrough depending on the retardation factor of the offspring species relative to the parent species. For the matrix diffusion model or other retention models with a long tail in the retention time distribution, this bias may be significant.

MARFA uses the extension to the time-domain particle tracking algorithm described by Painter [2008]. In this algorithm, decay and the resulting transformation to the next species in the decay chain are simulated as random events that transform the particle's entire mass to the offspring species; the particle's mass does not change until/unless the decay event occurs. If a decay event occurs in a segment, the total residence time in the segment is calculated as a combination of sampled residence times for the parent and offspring species.

To be more specific, suppose that a particle enters a segment as species A. Species A decays to species B according to the first-order decay law with a given decay constant. A decay time t_d is first sampled for species A. Two residence times denoted t_A and t_B for

species A and B, respectively, are then sampled. (The residence time is the sum of the groundwater travel time and the retention time.) If t_A is less than the decay time, the particle survives the segment as species A, the particle's clock is advanced by t_A, and the algorithm proceeds to the next segment.

If t_A is greater than the decay time, the particle decays in the segment, and the clock is advanced by the amount $t_d + \left(1 - \frac{t_d}{t_A}\right) t_B$. The first term in this expression represents the time in the segment spent as species A, and the second term represents the time spent as species B.

An important detail to note is that the sampled full-segment residence times t_A and t_B must be perfectly correlated for the algorithm to work properly. Algorithmically, perfect correlation is enforced by using the same random number when generating a sample of t_A and t_B. It is easy to show that perfect correlation is required for the special case when A and B have identical sorption properties. Numerical experiments confirm this requirement in general; sampling the residence times for the parent and offspring species independently results in censoring of the residence-time distribution for any retention model with strong kinetic controls, thereby shifting the breakthrough curves to earlier times. The physical reason for this censoring is that a new sampling implies starting the offspring nuclide in the mobile fluid, whereas the parent nuclide is much more likely to be in the immobilized state (i.e., somewhere in the matrix) when the decay event occurs.

For clarity, the algorithm just described is for a two member chain. The algorithm can be applied recursively, thus allowing decay chains of arbitrary length and multiple decay events in a single segment.

2.5 Pathways with Time-Dependent Velocity or Retention Parameters

Given the long time frames considered in assessments of potential high-level nuclear waste repositories in Sweden and Finland, significant changes in groundwater velocity are expected under changing future climate states. Future flow conditions are typically represented in performance assessment studies as piecewise steady; the same approach is taken in MARFA. That is, a sequence of steady flow fields is used with abrupt changes from one steady flow field to the next at specified times. A similar strategy is used to represent changing sorption properties. This capability was new to MARFA Version 3.2.

Glaciation modeling cases are specific examples of transient flow fields that need be accommodated in performance assessment studies. In current conceptualizations [Jaquet and Siegel 2006], the repository experiences a brief high-flow "flushing" period as the edge of the advancing ice sheet passes over the repository. A similar flushing period occurs during retreat of the ice sheet. In between the two flushing periods, alternative modeling cases of potential interest include a return to the preglacial flow or greatly reduced flow relative to non-glacial conditions because of reduced groundwater recharge. In both modeling cases, salinity is expected to be decreased and redox conditions changed from reducing to oxidizing during the glacial period, with

potentially large reduction in distribution coefficients for some radionuclides [Crawford et al. 2006]. Glaciation modeling cases are demanding numerically because of the potentially large changes in flow velocity and the potential for very low velocities during the nonflushing glacial period.

MARFA uses special handling of the residence time if the particle experiences a flow change while in transit. Specifically, upon reaching the time of a stepwise change in flow velocity, a depth *z* in the matrix is sampled. This value represents the depth in the matrix at the time of the flow change. The probability distribution for this depth depends on the current time (relative to the time at which the particle enters the matrix) and the matrix diffusion parameters. Given a sample z value, the time required to diffuse back to the fracture can then be calculated and sampled. Once the particle returns to the fracture, the existing residence time distribution using the new flow velocity may be sampled to account for the time required to finish the segment.

The distribution for the return time depends on whether the unlimited or limited diffusion model is used as the retention model. If the tabular option is specified for the retention model, then the procedure is the same as for the limited diffusion model but with effective parameters for the matrix properties.

2.5.1 Probability Density for Depth in the Matrix

The probability density for the random distance z in the matrix at time *t* (relative to the start of the segment) is needed. Using the equivalence between probability density and concentration, the required density function can be calculated by considering advective flow in the fracture with diffusion in the adjacent matrix. The appropriate initial condition for the problem is a Dirac-δ function located at the fracture inlet (equivalent to pulse input at $t = 0$). The concentration in the matrix for this modeling case is given by Sudicky and Frind [1984].

A simpler approximation is to ignore advection and use a pure-diffusion model to calculate the probability density for location in the matrix. The appropriate initial condition is a Dirac-δ located at z=0. For the case of unlimited diffusion, the result is given by Carslaw and Jaeger [1959] as

$$f(z) = \frac{G}{2\sqrt{\pi t}} \exp\left[-\frac{(Gz)^2}{4t} \right] \tag{9}$$

where t [T] is time, z [L] is distance perpendicular to the fracture, and $G \equiv \sqrt{R/D_{eff}}$ [$T^{1/2}$/L], R [-] is retardation factor in the matrix, and D_{eff} [L^2/T] is matrix effective diffusion coefficient. Equation 9 has been compared with the more rigorous expression of Sudicky and Frind [1984], and in all cases the agreement is very close. The simpler expression of Eq. (9) is used in MARFA.

In the limited diffusion model, diffusion is into a limited zone of width Δ in the matrix. To calculate the penetration depth, consider slab geometry with no-flow boundary at both sides (z = Δ, and z = 0). The initial concentration is a Dirac-δ located at z = 0.

Carslaw and Jaeger [1959] provide results for this configuration, which upon inserting the δ-function initial condition simplifies to

$$f(z) = \frac{1}{\Delta}\left\{1 + 2\sum_{n=1}^{\infty} \exp\left[-n^2\pi^2 T\right]\cos(n\pi\zeta)\right\} \tag{10}$$

where $\zeta \equiv z/\Delta$ and $T \equiv \frac{t}{t_0}$ [-] is a dimensionless time with $t_0 \equiv \frac{R\,\theta\Delta^2}{D_{eff}}$ [T]. Note that the characteristic time t_0 is directly related to the two MARFA input parameters $\kappa = \sqrt{\theta D_{eff} R}$ and $\eta = \frac{\Delta}{D_{eff}}$ as $t_0 = (\kappa\eta)^2$.

2.5.2 Probability Density for Time of Return

Given a sample z^* from the distribution defined by Eqs. (9) or (10), the time required to diffuse back to the fracture is now required. The distribution of return times may be calculated from diffusion in a semi-infinite domain with zero concentration boundary at z=0 (representing the fracture); the appropriate initial condition is a Dirac-δ function located at z^*, the particle's location at the time of the flow change. The concentration is given by [Carslaw and Jaeger 1959]

$$C(z,t) = \frac{G}{2\sqrt{\pi\, t}}\left\{\exp\left[-(z-z^*)^2 G^2/4\,t\right] - \exp\left[-(z+z^*)^2 G^2/4\,t\right]\right\} \tag{11}$$

Here t is defined relative to the time at which the flow change occurs.

The mass flux into the fracture is given by $-\frac{D_{eff}}{R}\frac{\partial C}{\partial z}\Big|_{z=0}$. Performing this calculation and interpreting the mass flux as the density for a conditional arrival time distribution gives

$$f(t\,|\,z^*) = \frac{G z^*}{2\sqrt{\pi\, t^3}}\exp\left(-(G z^*)^2/4\,t\right) \tag{12}$$

The corresponding expression for the limited diffusion case is

$$f(t\,|\,z^*) = \frac{\pi}{t_0}\sum_{n=0}^{\infty}(2n+1)(-1)^n \exp\left[-(2n+1)^2\pi^2 T/4\right]\cos\left(\frac{(2n+1)\pi(1-\zeta^*)}{2}\right) \tag{13}$$

where $\zeta^* \equiv z^*/\Delta$.

2.6 Downscaling Algorithm

As discussed in Section 2, MARFA is designed to accept pathway trajectories from the CONNECTFLOW [Hartley and Holton 2003] flow modeling software. Permeability fields for CONNECTFLOW simulations may be DFN representations, CPM equivalents upscaled from DFN models, or nested models combining DFN and CPM representations. With upscaled CPM representations, subgrid velocity fluctuations have been averaged away in the homogenization process. Averaging is considered appropriate for regional scale flow modeling, but is questionable for transport calculations that may be sensitive to local (subgrid) velocity variability.

MARFA has an option to use a unique downscaling algorithm to restore the lost transport effects of subgrid velocity variability. Details were presented by Painter and Cvetkovic [2005], where the method was presented as streamline extrapolation and described as transport "upscaling". In MARFA, the same algorithm is used to simulate lost subgrid velocity variability in an upscaled CPM flow model; in that sense, it is more appropriately described as transport "downscaling".

The downscaling algorithm uses a set of pure-advection trajectories extracted from CONNECTFLOW DFN flow simulations that are approximately the size of the grid cells in the CPM flow model. Those trajectories are referred to as "subgrid trajectories." If the optional downscaling mode is used, the first step in a MARFA simulation is to decompose the subgrid trajectories into segments, where a segment is defined as the part of a trajectory that passes through a single fracture. Each segment has associated with it values for τ, β, and length. When MARFA is using pathway trajectories calculated from a CPM flow model and downscaling is activated, the segment pool is sampled randomly to provide the subgrid velocity variability. Specifically, each time a particle is advanced along the CPM trajectory, a segment with its associated properties is first drawn from the segment pool.[3] The τ and β values for that segment are then scaled according to the local value of the hydraulic gradient, as described below, and the particle is moved on that segment using the algorithms described in Sections 3.1, 3.4 and 3.5.

The τ and β values sampled from the segment pool are modified before they are used. This modification is done because the hydraulic gradient that is imposed on the DFN when generating the subgrid trajectories is, in general, not the same as the local hydraulic gradient calculated by the regional flow model applicable at the given point on the pathway. MARFA requires that a local hydraulic gradient as calculated from the CPM flow model be specified at each point along the CPM trajectory. Similarly, the hydraulic gradient used in generating each subgrid trajectory set must also be specified. The sampled τ and β values are scaled by the ratio of the two gradients before the particle is advanced.

[3]In its simplest form, the sampling algorithm is purely random. More generally, the algorithm incorporates sequential correlation (persistence) in the sequence. Thus, if a high-velocity segment is sampled, it is more likely a high-velocity segment will be sampled the following step, consistent with analyses of DFN data [Painter and Cvetkovic, 2005]. Details of the sampling algorithm can be found in Painter and Cvetkovic [2005].

An additional complication arises when the underlying DFN is anisotropic. In that situation, the segment statistics depend on the direction of the applied hydraulic gradient. To use the downscaling algorithm with anisotropic DFNs, the user must provide multiple sets of subgrid trajectories, with each set corresponding to a different direction for the applied gradient. Each time the segment sampling is performed, MARFA first determines which of the applied gradients is most closely aligned with the local direction of the CPM pathway trajectory and then uses that subgrid trajectory set.

2.7 Reconstruction of the Breakthrough Curves

The Monte Carlo algorithms produce output in the form of particle arrival times at pathway end points. Cumulative mass discharge (cumulative breakthrough) at a given time can be readily constructed from the arrival times by simply identifying the amount of mass arriving before the specified time. This procedure is equivalent to estimating cumulative probability distributions from a set of random samples. Estimating the mass discharge rate (instantaneous breakthrough) is more difficult. Fortunately, the task of reconstructing breakthrough curves from the particle arrival times is analogous to reconstructing a probability density from a set of sampled values, a classical problem in statistics with a large literature [e.g., Silverman 1986].

An adaptive kernel method that adjusts the degree of smoothing to the local data density is used in MARFA. The adaptive kernel methods are well-known algorithms that have been investigated in detail [Silverman 1986]. In these algorithms, a nearest neighbor method is first used to estimate an initial or "pilot" estimate. The pilot estimate is then used to calculate a variable kernel width for use in a kernel estimation method. The adaptive two-step method is appealing because of its capability to provide stable and smooth density estimations for a wide variety of data distributions, while maintaining sensitivity to local details. MARFA uses a lognormal kernel. This kernel has the advantage of always producing zero breakthrough estimates for non-positive times.

The pilot estimate for the breakthrough curve is obtained at a specified time t from a generalized nearest neighbor estimator of order k

$$f'_{ar}(t) = \frac{1}{W}\sum_{i=1}^{n} w_i K\left(t, t_i, d_k\right) \tag{14}$$

where $t_1, t_2, \cdots, t_n$ are the arrival times, $w_1, w_2, \ldots, w_n$ are the statistical weights (masses) for the individual packets, $W = \sum_{i=1}^{n} w_i$, d_k is the distance from t to the k-th nearest arrival time, $K\left(t, t_i, d_k\right)$ is the kernel (lognormal density function with geometric mean t and log-standard deviation d_k). MARFA uses k = min (25, n); results are not sensitive to this value.

Once the pilot estimate is known, the estimate for the breakthrough curve is obtained as

$$\hat{f}_{ar}(t) = \sum_{i=1}^{n} w_i K(t, t_i, h\omega_i) \tag{15}$$

where $\omega_i = \left[f'_{ar}(t_i) / \gamma \right]^{-1/2}$, γ is the geometric mean of the $f'_{ar}(t_i)$, and h is the global bandwidth. The product $h\omega_i$ is a local bandwidth that adapts the kernel width according to the local data density. For nearly normal data, a good estimate for the optimal bandwidth is given by Silverman [1986] as

$$h_{opt} = 0.9\, A\, n^{-\nu} \tag{16}$$

where h_{opt} is the optimal bandwidth, A is the smaller of the standard deviation and $(Q_{75} - Q_{25})/1.34$, Q_p is the p-th percentile, n is number of data points, and ν is a bandwidth sensitivity parameter. The value $\nu = 0.2$ is optimum for smoothly varying breakthrough curves [Silverman, 1986] and is recommended for modeling cases that have steady-state flow; numerical experiments suggest that slightly larger values (for example, 0.3–0.5) are better for modeling cases involving abrupt flow changes. The bandwidth parameter was hardwired to the value 0.2 in MARFA Version 3.1, but is an input parameter in MARFA Version 3.2 and later versions.

3 MARFA 3.3.1 ALGORITHMS

MARFA 3.2.3 and 3.3.1 use many of the same algorithms and share much of the same code base. This section describes only those algorithms that are unique to version 3.3.1.

3.1 Node Routing

MARFA 3.3.1 does not use fixed transport pathways. Instead, particles are routed dynamically through a node network derived from a CONNECTFLOW or other flow field. The node network is specified by a CONNECTFLOW PTH file. The PTH file provides for each CONNECTFLOW transport node a list of downstream neighbors, probability of traveling to each neighbor, and properties of each downstream link. This information is provided in packed arrays in the CONNECTFLOW PTH file. MARFA 3.3.1 uses a Fortran 90/95-derived data type called *node* to store the same information. Thus, the multiple packed arrays of the CONNECTFLOW PTH format need to be converted to a single array of type *node* for use in MARFA 3.3.1. Each element in the array contains the information for one node, which is sufficient to route to the next node. The advantage of the derived data type is that it leads to a very compact implementation of the node-routing algorithm. A utility routine that reads PTH files and creates the MARFA node data input file has been developed and is controlled with MARFA 3.3.1.

The logic for the particle routing follows.

(1) Select an initial node. This is done by uniformly sampling from a predefined list of nodes associated with radionuclide sources. From the selected node, a particle is launched.

(2) Select a downstream node. A downstream node is selected based on the probability of traveling to the downstream neighbors for the current node.

(3) Evaluate the transport time. The advection and retention times for transitioning to the downstream node are calculated as in previous versions.

(4) Advance the particle to the next node, and advance the clock based on the sampled transit time.

Steps 2–4 are repeated until the particle passes a flow boundary, reaches the end of the decay chain, or becomes stuck or stranded (defined later in this document). Particle decay and ingrowth are handled as in previous versions.

This approach easily accommodates a change in flow direction because a check can be made at Step 2 to determine which flow field (i.e., which node network) is active at the time. If a change in flow direction is encountered while a particle is in transit between two nodes, the particle is advanced to the downstream node before any changes in direction or retention properties are applied. This introduces an approximation into the transport calculation. However, with sufficient spatial resolution, the impact should be small.

Three additional issues arise in the use of node routing: stuck particles, stranded particles, and node renumbering.

3.2 Stuck Particles

Stuck particles are particles that become trapped in a series of nodes that form a loop. Loops can be closed, meaning once a particle enters the loop, the particle becomes permanently trapped in the loop with probability one. Loops can also be open; once a particle enters an open loop, the particle has some non-zero probability of exiting the loop. Loops occur because of inaccuracy in the finite-element solution to the flow field.

MARFA 3.3.1 identifies a stuck particle by recording the number of visits a particle makes to each node. When the number of visits exceeds a user-defined threshold, a particle is declared stuck and terminated. The stuck particles are recorded in the particle breakthrough file *results.dat*, but they are identified as stuck and are filtered out by default by the breakthrough reconstruction algorithm. The number of visits allowed is a user-defined parameter.

MARFA 3.3.1 reports statistics on the number of particles experiencing loops and the number of particles terminated due to excessive looping. These results are written to a file named *loop_diagnosis.rlt* and are intended to help identify unreliable results caused by an inaccurate flow field as input.

3.3 Stranded Particles

In addition to particles becoming stuck, particles can become stranded. Stranded particles can arise during the evaluation of transport with flow changes. If a particle resides on a node prior to a flow change and if this node has no downstream neighbors defined after the flow change, the particle has become stranded. This occurs when differing regions of the fracture network are nearly stagnant under differing flow directions. Similar to stuck particles, stranded particles are terminated and are recorded in the particle breakthrough file *results.dat*. These particles are identified as stranded and are automatically filtered out by the breakthrough reconstruction algorithm.

3.4 Node Renumbering

MARFA 3.3.1 requires a node data file for each flow period. Because of a potential to have nodes numbered differently among node data sets, a translation table is provided for each flow change to ensure that nodes are properly mapped between consecutive data sets. Node renumbering is not expected to be an issue for repository safety assessment calculations, but may be for some studies using generic geometries.

4 INSTALLING AND RUNNING MARFA

MARFA is written in Fortran 95. A Fortran compiler that supports the allocatable component extension [ISO/IEC 1998] is required to compile the source code. MARFA is otherwise platform independent. A generic *makefile* is included with MARFA. This *makefile* will work unmodified with the gfortran on most systems. To compile with another compiler, edit the *makefile* and replace the Fortran 95 compile/link commands with the one appropriate for the local system. Compiler flags should also be replaced with appropriate values.

MARFA reads six files that contain data required for efficient sampling of retention time distributions. These are included in the directory named *data*. The required data files must be placed in the same directory, which can be placed in any accessible location on the local file system. The local file system path to the directory is part of the required input (See Appendix A).

A complete set of input for one example simulation is included with the MARFA installation package. This example is useful as an installation test and is described in Section 6.

MARFA does not have a graphical interface. It is executed from the command line. All input comes from a series of ASCII files, as described in Appendix A. Warning, error and other diagnostic messages are written to the screen and may be redirected to a file, if desired. Three output files are created in the run directory. One file contains particle arrival times and the other two contain reconstructed breakthrough curves on a mass or activity basis. Details of the output are described in Appendix B.

The default operation of MARFA (i.e., without a command line argument) results in the complete execution of the MARFA code, including particle transport calculations and breakthrough curve calculations. MARFA can also skip the particle transport calculation and reconstruct breakthrough curves from existing particle arrival times stored in the file *results.dat*. This option is activated using the command line argument ***ppo*** (post-processor only) by typing "marfa ppo."

5 VERIFICATION TESTS

Seven verification tests were developed for the MARFA software. These tests are *software verification tests* designed to verify that the software solves the underlying mathematical equations representing radionuclide transport. Model support or validation – the process of accumulating evidence to support the validity of the underlying mathematical representation of transport – is beyond the scope of this document.

All the verification tests use a two-member decay chain $A \rightarrow B$. The half-life is 10,000 years for both species, unless otherwise noted. Radionuclides are introduced into the system as species A. The source strength is initially 0.001 mol/yr, and decreases exponentially with decay constant of 0.001 yr^{-1}.

5.1 Unlimited Diffusion

Test 1 used a pathway with a single segment of length $\ell = 100$ m and the unlimited matrix diffusion model. The governing equations for this system are

$$\frac{\partial C_i}{\partial t} + v\frac{\partial C_i}{\partial x} - \alpha|v|\frac{\partial^2 C_i}{\partial x^2} = \frac{D_{eff}}{b(x)}\frac{\partial M_i}{\partial z}\bigg|_{z=0} - \lambda_i C_i + \lambda_{i-1} C_{i-1} \tag{17a}$$

$$R_i \frac{\partial M_i}{\partial t} = \frac{D_{eff}}{\theta}\frac{\partial^2 M_i}{\partial z^2} - \lambda_i R_i M_i + \lambda_{i-1} R_{i-1} M_{i-1} \tag{17b}$$

for *i=1,2* and with boundary/initial conditions $C_i(t,x) = M_i(t,x,0)$, $M_i(t,x,\infty) = 0$, $C_1(t,0) = f(t)$, $C_2(t,0) = 0$, and $C_i(0,x) = 0$.

Here $C_i(t,x)$ is the concentration in the mobile pathway (fracture) for the *i*-th member of the chain, $M_i(t,x,z)$ is the concentration in the matrix, R_i is the matrix retardation factor, x is distance in the direction of the pathway, z is distance orthogonal to the pathway, and $C_0 \equiv 0$. The system was solved by discretizing the z-direction using block-centered finite differences with 18 grid blocks. That procedure eliminated the z-dependence and resulted in a total of 38 coupled two-dimensional (x,t) equations, which were solved using the method of lines.

Mass breakthrough curves for the matrix-diffusion verification test are shown in Figure 2. The species-independent parameters are $v = 1$ m/yr, $\alpha = 0.5$ m, $b = 0.1$ mm, $\theta = 0.01$ and $D_{eff} = 10^{-8}$ m^2/yr. The matrix retardation factors are 1,000 and 100 for species A and B, respectively. The MARFA results agree very well with the target benchmark solution over a wide range and show discernible differences only at the extreme leading and trailing edges of the breakthrough curve. The MARFA results are uncertain in the tails of the breakthrough curve because a finite number of particles are used. The numerical results are also suspect there because of numerical dispersion. Other tests were performed using different combinations for parameters (not shown), and all of these resulted in similar good agreement.

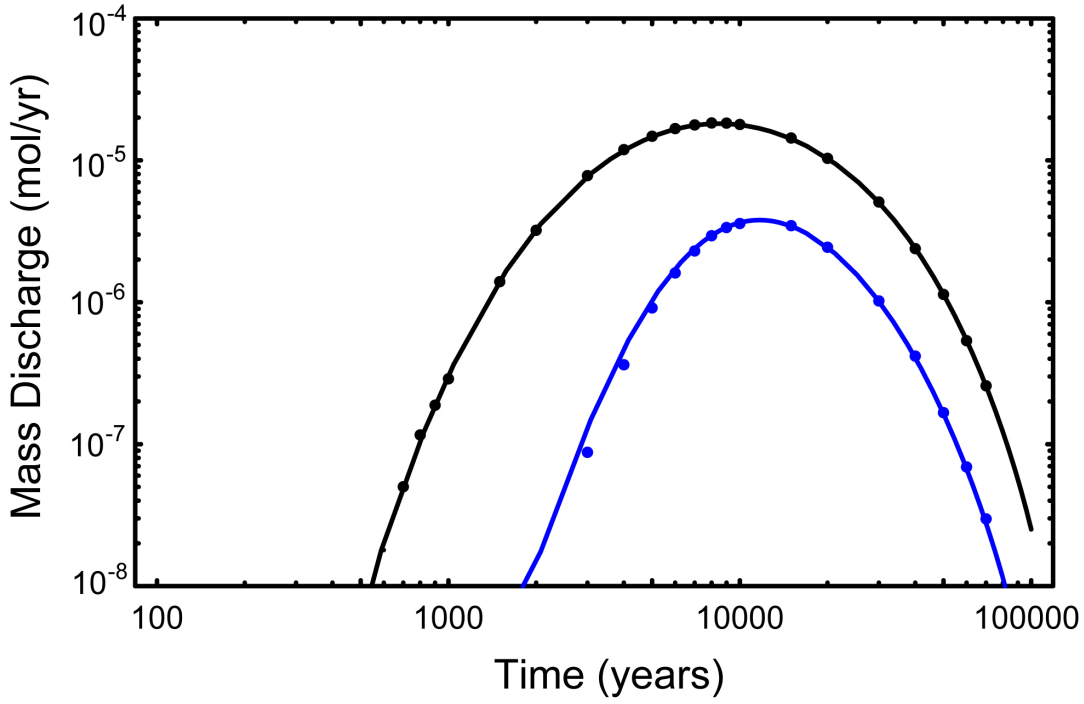

Figure 2. *Results of Verification Test 1. Solid curves are benchmark solution for the mass discharge. Individual data points are mass discharge from MARFA. The blue lines and points are for species A. Black represents species B.*

5.2 Limited Diffusion

Test 2 differs from Test 1 in two respects. First, particles are only allowed to diffuse into a 4.75 mm region matrix adjacent to the fracture. Thus, the boundary condition $M_i(t,x,\infty)=0$ from Test 1 is replaced with $\left.\frac{\partial M_i(t,x,z)}{\partial z}\right|_{z=\Delta}=0$, where Δ = 4.75 mm. Second, the matrix retardation factors are 200 and 500 for species A and B, respectively. Other parameters are as in Test 1.

Results of Test 2 are shown in Figure 3. The MARFA results agree very well with the benchmark solution over the entire range.

Test 2a is a slight variant of Test 2. In Test 2a, the 100 m transport path is sub-divided into 10 segments of length 10 m each. Results of Test 2a (not shown) are identical to those of Test 2, thus confirming that a segmented pathway is correctly handled in MARFA.

5.3 Multiple Pathways

Test 3 is designed to test the capability to simulate multiple sources and multiple pathways. The test has two pathways originating from two sources. Pathway one is identical to the pathway in Test 1, while pathway 2 is identical to that of Test 2. The same amount of mass is released from the two sources. The instantaneous breakthrough curves for Test 3 are shown in Figure 4. The MARFA results agree very well with the benchmark solution over the entire range.

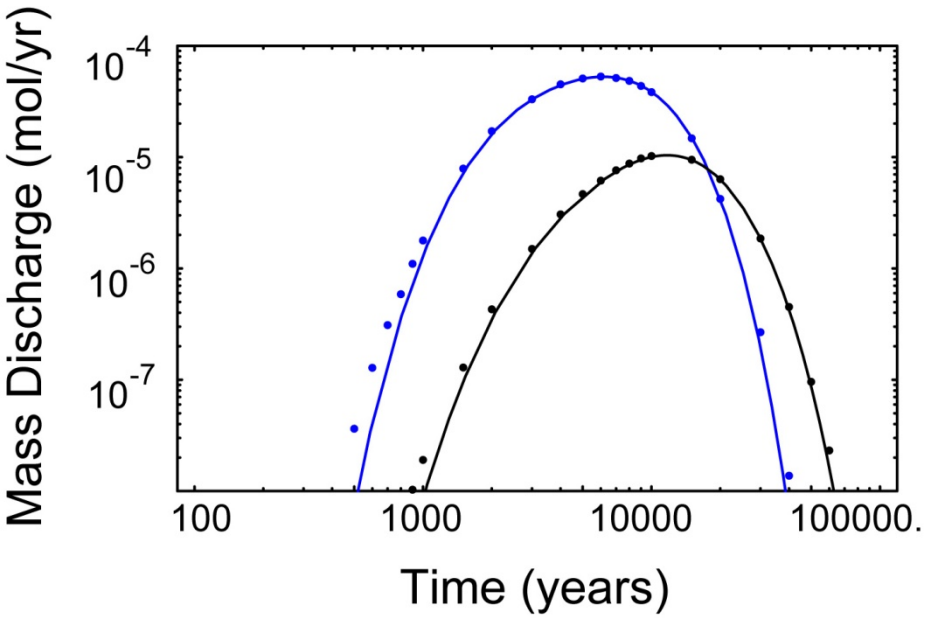

Figure 3. *Results of Verification Test 2. Solid curves are benchmark solution for the mass discharge. Individual data points are mass discharge from MARFA. The blue lines and points are for species A. Black represents species B.*

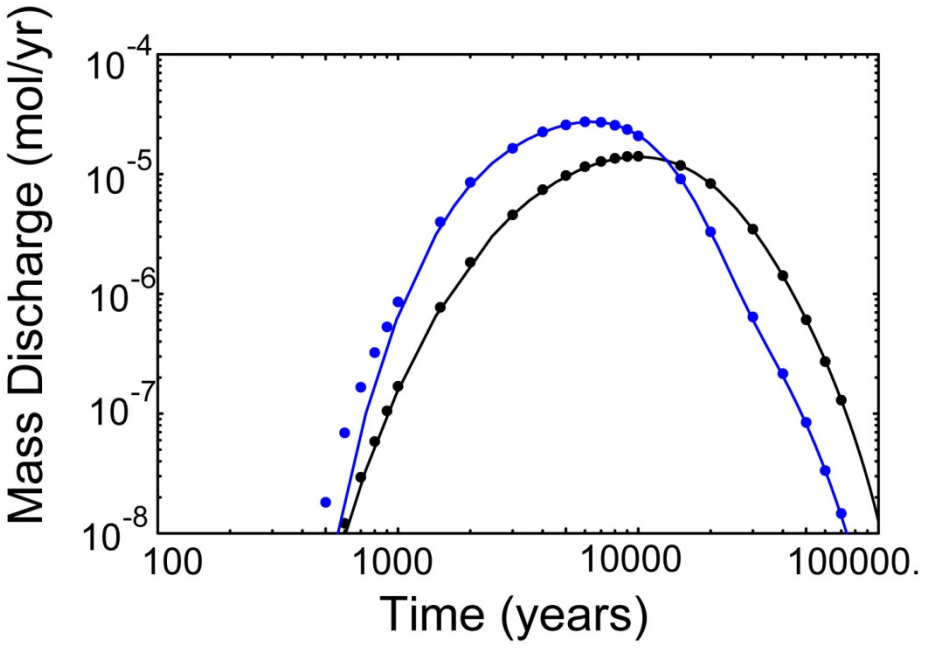

Figure 4. *Results of Verification Test 3. Solid curves are benchmark solution for the mass discharge. Individual data points are mass discharge from MARFA. The blue lines and points are for species A. Black represents species B.*

5.4 Low Flow Period with Unlimited Diffusion

Test 4 is designed to test the new capabilities to model changing flow velocities. The flow velocity in this test is initially 1 m/yr. At t = 2,000 yrs, the flow is changed to

0.01 m/yr, representing a glacial nonflushing period. At t = 7,000 yrs, the flow returns to 1 m/yr. The sorption properties are unchanged in this test. The matrix retardation factor for species A is 500. Other parameters are identical to Test 1. Results of this test are shown in Figure 5. The MARFA results agree with the benchmark solution over the entire range, except for statistical fluctuations at very low breakthrough values.

5.5 Flow Period with Limited Diffusion

Test 5 uses the same piecewise constant-in-time velocity as Test 4 but with the limited diffusion model (4.75 mm matrix). Other parameters are identical to Test 2. As in Test 2, two variants are considered: one with a single segment representing the pathway, and one with the pathway composed of 10 segments. Results for the single pathway and multiple pathway variants are shown in Figures 6a and 6b, respectively. The MARFA mass discharge results are slightly larger than the benchmark solution immediately following the second flow change in the single pathway variant. This deviation is caused by the particles that encounter both a flow change and a decay event when in the segment, a situation that is handled approximately in MARFA. When the pathway is discretized into multiple segments, a particle is less likely to experience both events in the same segment and the MARFA results closely approximate the benchmark solution over the entire range.

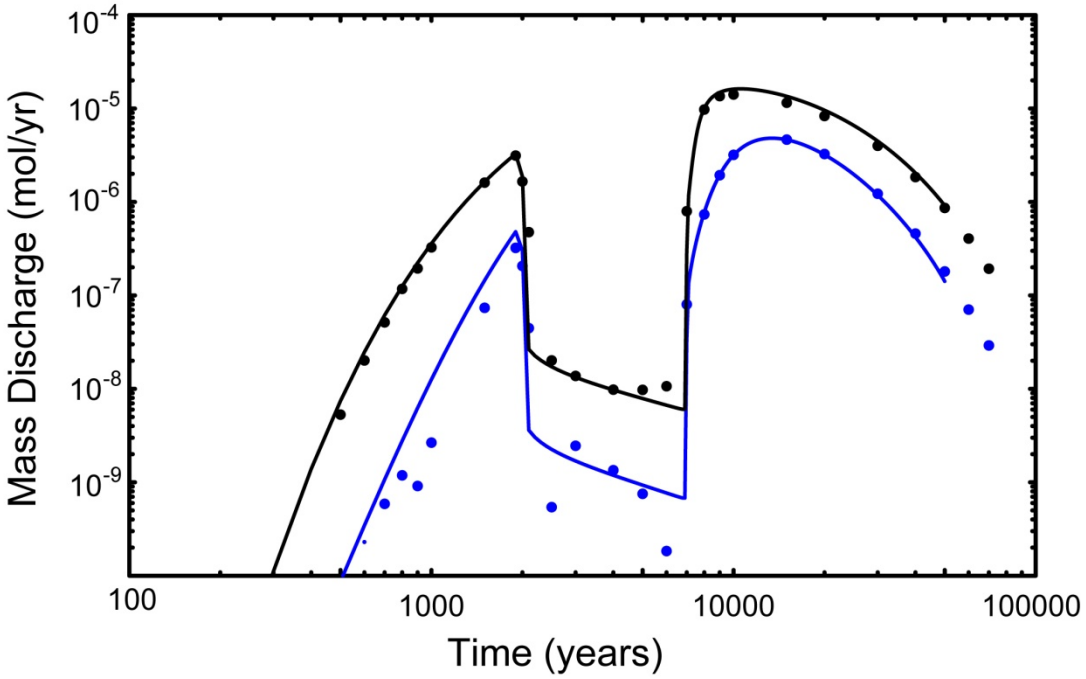

Figure 5. *Results of Verification Test 4. Solid curves are benchmark solution for the mass discharge. Individual data points are mass discharge from MARFA. The blue lines and points are for species A. Black represents species B.*

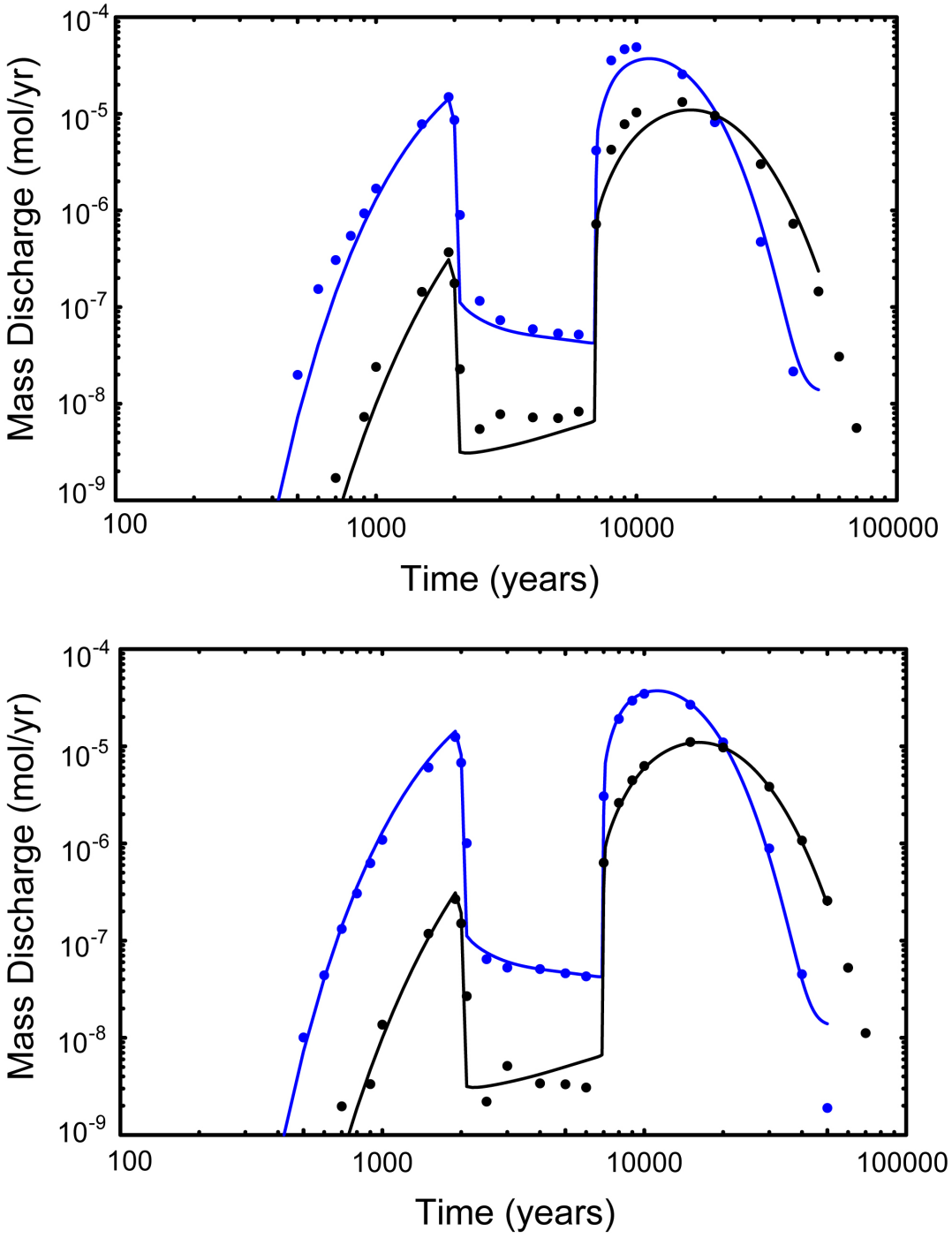

Figure 6. *Results of Verification Test 5 using a single- (top) or multiple-segment pathway. Solid curves are benchmark solution for the mass discharge. Individual data points are mass discharge from MARFA. The blue lines and points are for species A. Black represents species B.*

5.6 High Flow Period with Unlimited Diffusion

In Test 6 the velocity increases by a factor of 10 in the period 2,000 to 7,000 years, consistent with a glacial flushing period. The matrix retardation factors are 200 and 200, for species A and B, respectively. Otherwise, the test is identical to Test 1. Results are shown in Figure 7. For these conditions, the MARFA results are slightly lower than the benchmark results for species B just after the end of the glacial flushing period. This period of deviation is brief. Moreover, the discharge during the flushing period, which is the risk significant quantity, is well approximated.

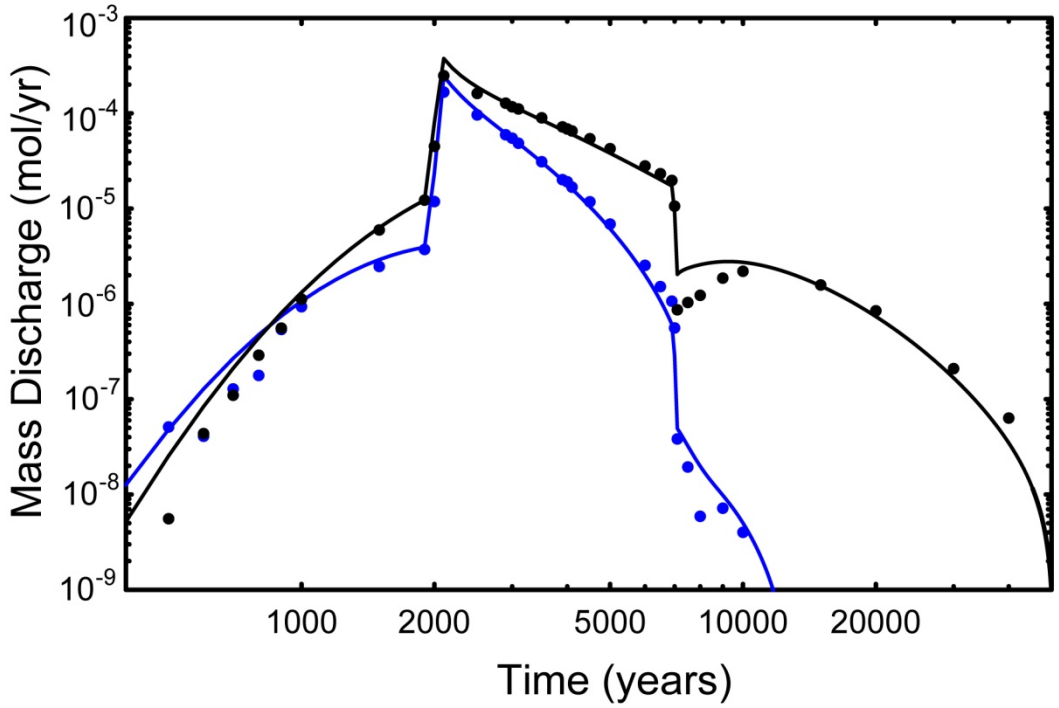

Figure 7. *Results of Verification Test 6. Solid curves are benchmark solution for the mass discharge. Individual data points are mass discharge from MARFA. The blue lines and points are for species A. Black represents species B.*

5.7 Changes in Flow and Sorption Properties

Test 7 uses the same flow history as Tests 4 and 5 and a limited diffusion model (4.75 mm matrix). The matrix retardation factors in Test 7 are 500 and 100 for species A and B, respectively, and are reduced by a factor of 10 during the glacial nonflushing period. Results are shown in Figure 8.

5.8 Limited Matrix Diffusion in Tabular Form

Test 8 is a limited diffusion scenario similar to Test 2 but implemented using the tabular retention time distribution option, which was introduced in Version 3.2.3. Diffusion is into a 4.75 cm matrix. The transport path is 10 m long and has a groundwater velocity of $v = 1$ m/yr and a dispersivity of $\alpha = 0.5$ m. The Mathematica script ThreeLayerDiffusion, which is a helper application in MARFA version 3.2.3, was used to

create the tabular retention time distribution. The three matrix layers were given identical properties: $b = 0.1$ mm, θ = 0.01, $D_{eff} = 10^{-6}$ m^2/yr, and matrix retardation factors of 200 and 500 for species A and B, respectively. Results are shown in Figure 9.

5.9 Diffusion into a Three-layer Matrix

Test 9 is a limited diffusion scenario similar to Test 8 but with a three-layer matrix with different diffusion coefficients and porosities among the layers. The first layer is 2.5 mm thick, the second layer is 4 mm and the third layer is 5 mm. The effective diffusion coefficients are 1×10^{-13} m^2/s, 1×10^{-12} m^2/s and 6×10^{-14} m^2/s for layers 1, 2 and 3, respectively. The porosities are 0.01, 0.05, and 0.005 for layers 1, 2, and 3, respectively. The transport path is 10 m long and has an initial groundwater velocity of $v = 1$ m/yr and a dispersivity of $\alpha = 0.5$ m. The Mathematica script ThreeLayerDiffusion was used to create the tabular retention time distribution. The aperture $b = 0.1$ mm and retardation factors (200 and 500 for species A and B, respectively) are identical for the three layers.

Three variants are considered: Test 9 has constant velocity. In Test 9a, the groundwater velocity drops to 0.01 m/yr in the period 2000 to 7000 years. In Test 9b, the groundwater velocity increases to 10 m/yr in the period 5000 to 7000 years. Results are shown in Figure 10.

5.10 Particle Splitting

Test 10 addresses new capability in Version 3.2.3 to split particles after a decay event. Splitting particles is a variance-reduction strategy. It attempts to reduce statistical uncertainty in the estimated breakthrough curve without commensurate increase in computational effort. Test 10 is identical to Test 1 except that the half-life for species A in Test 10 is 1 million years. In addition, particles are split 50 to 1 after a decay event.

Results for species B are shown in Figure 11. Because of the long half-life for species A, a relatively small number of decay events occur, which makes the estimated breakthrough for species B noisy. Splitting particles after a decay event reduces the statistical variance in the estimated breakthrough for species B. As can be seen in Figure 11, the splitting procedure does not produce bias in the result.

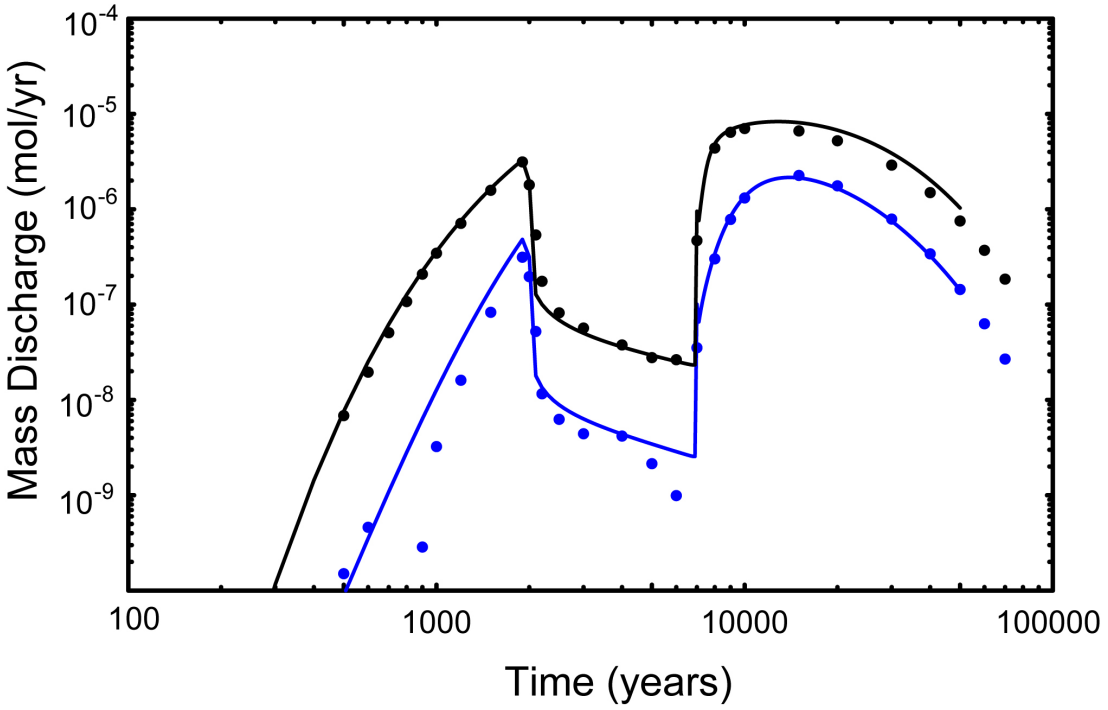

Figure 8. *Results of Verification Test 7. Solid curves are benchmark solution for the mass discharge. Individual data points are mass discharge from MARFA. The blue lines and points are for species A. Black represents species B.*

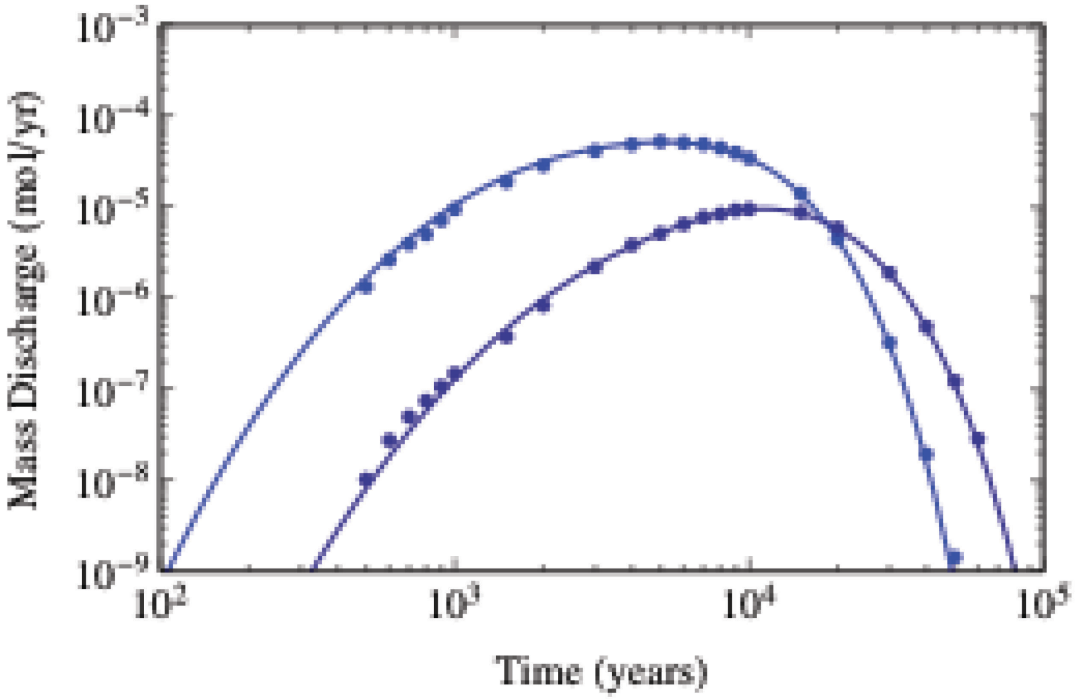

Figure 9. *Results of Verification Test 8. Solid curves are benchmark solution for the mass discharge. Individual data points are mass discharge from MARFA. The blue lines and points are for species A. Black represents species B.*

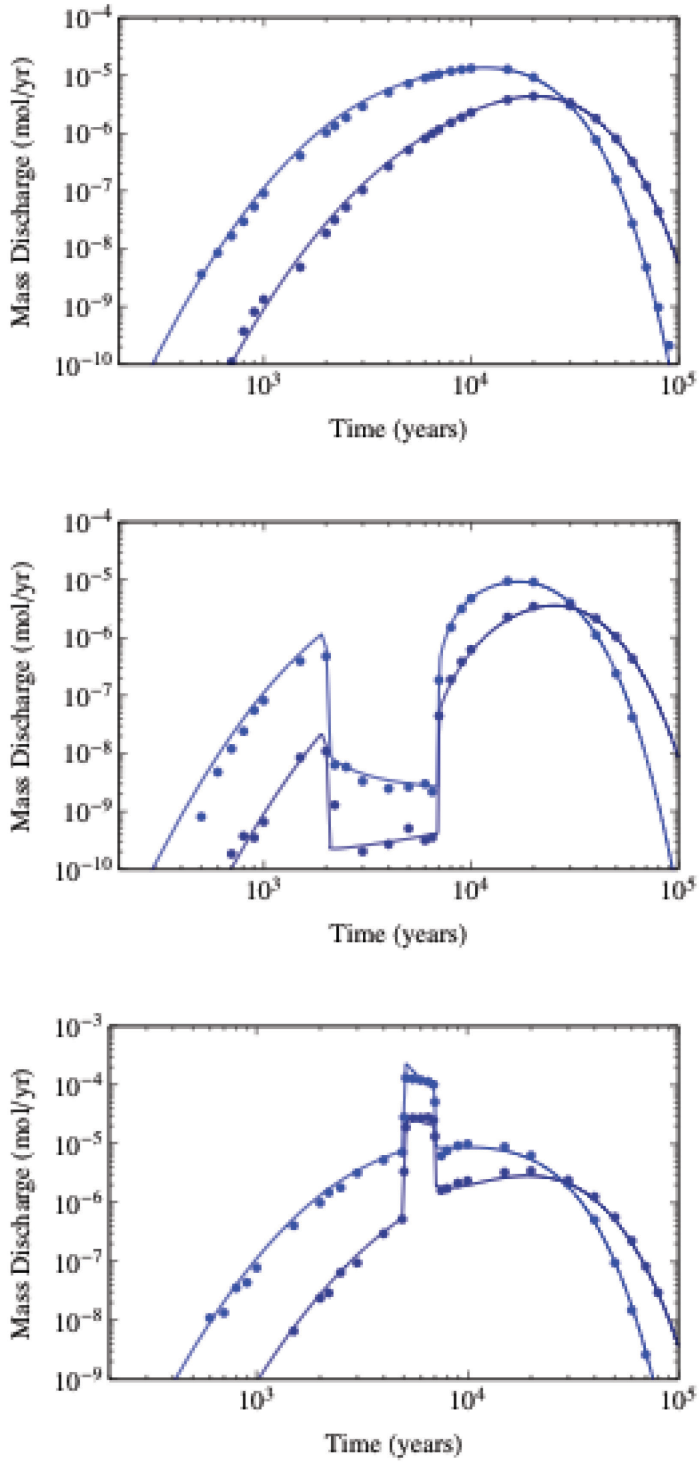

Figure 10. *Results of Verification Test 9. Solid curves are benchmark solution for the mass discharge. Individual data points are mass discharge from MARFA. The blue lines and points are for species A. Black represents species B.*

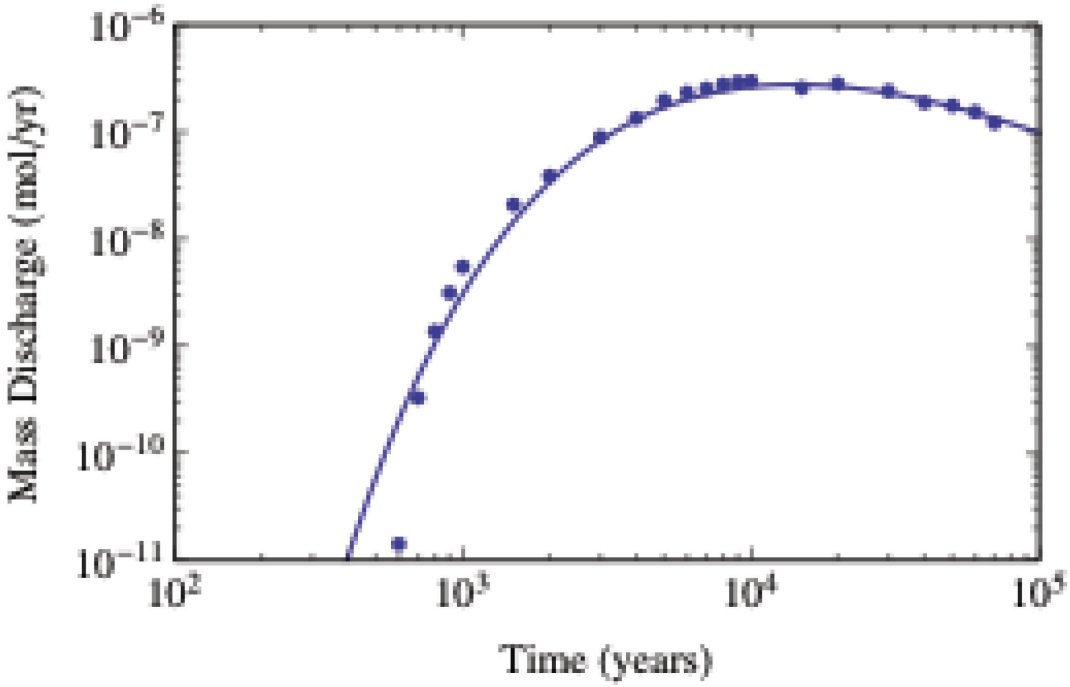

Figure 11. *Results of Verification Test 10. Solid curves are benchmark solution for the mass discharge. Individual data points are mass discharge from MARFA. Results for species B are shown.*

6 VERIFICATION TESTS FOR VERSION 3.3.1

Three verification tests were developed for the MARFA software. These tests are *software verification tests* designed to verify that the software solves the underlying mathematical equations representing radionuclide transport. Model support or validation—the process of accumulating evidence to support the validity of the underlying mathematical representation of transport—is beyond the scope of this document.

In all three tests, MARFA 3.3.1 was compared with the previously verified MARFA 3.2.3. These tests address (i) the retention models supported by MARFA 3.3.1 (unlimited diffusion, limited matrix diffusion, equilibrium sorption), (ii) the ability to evaluate changes in flow velocities, and (iii) the capability to evaluate multiple source terms and multiple pathways.

6.1 Tests of the Retention Model

The first set of tests used a two-member decay chain (A→B) along a one-dimensional pathway. The pathway was initially free of radionuclide mass; one mole of species A was instantaneously injected into the upstream end of the pathway to start the simulation. The transport pathway consisted of a 100-m pathway divided into 1-m segments. Node data files for MARFA 3.3.1 were constructed to represent the one-dimensional deterministic pathway.

Mass breakthrough curves for the matrix diffusion model are shown in Figure 12. For this analysis, the advective travel time is 100 years and the global transport resistance parameter for the pathway is 10^6 yr/m. The retention parameter κ was set to 0.003162 m/yr$^{-1/2}$ and 0.0001 m/yr$^{-1/2}$, respectively, for radionuclides *A* and *B*. The longitudinal dispersion length is 0.5 m. The half-life of both radionuclides *A* and *B* is 10,000 years. This result shows good agreement between MARFA 3.3.1 and MARFA 3.2.3.

Similar to the result shown for the matrix diffusion model, the results of the equilibrium sorption model and the limited matrix diffusion model tests show good agreement between MARFA 3.3.1 and MARFA 3.2.3 (results not shown). The results of these tests (Test1a, Test1b, Test1c) confirm that the matrix diffusion, limited matrix diffusion, and equilibrium sorption retention models are correctly implemented in MARFA 3.3.1.

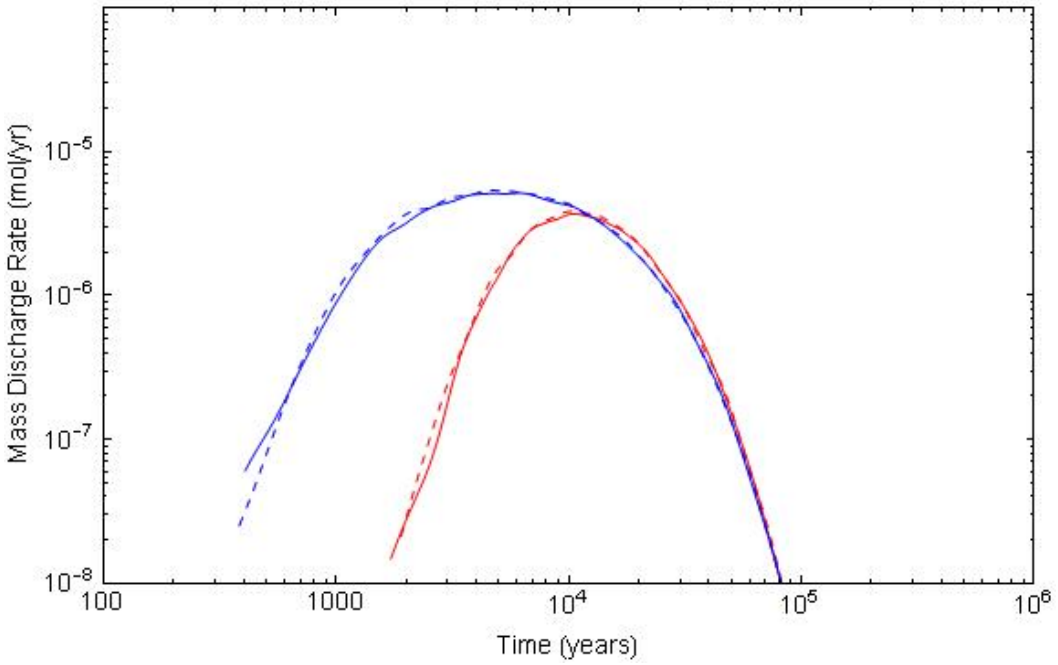

Figure 12. *Result of Version 3.3.1 Verification Test 1. The solid lines show the results of MARFA 3.2.3, and the dashed lines show the results of MARFA 3.3.1. The red lines are for radionuclide A, and the blue lines are for radionuclide B.*

6.2 Flow Change Test

This test was designed to assess the capability of MARFA 3.3.1 to model changing flow velocities. The flow velocity in this test is initially 1 m/yr. At 10,000 years, the flow velocity changes to 2 m/yr. The sorption properties are unchanged at the time of the velocity change. The source, the pathway, and the retention parameters are identical to those in Test1a (unlimited matrix diffusion test).

The results of this test are shown in Figure 13. These results show good agreement between MARFA 3.3.1 and MARFA 3.2.3.

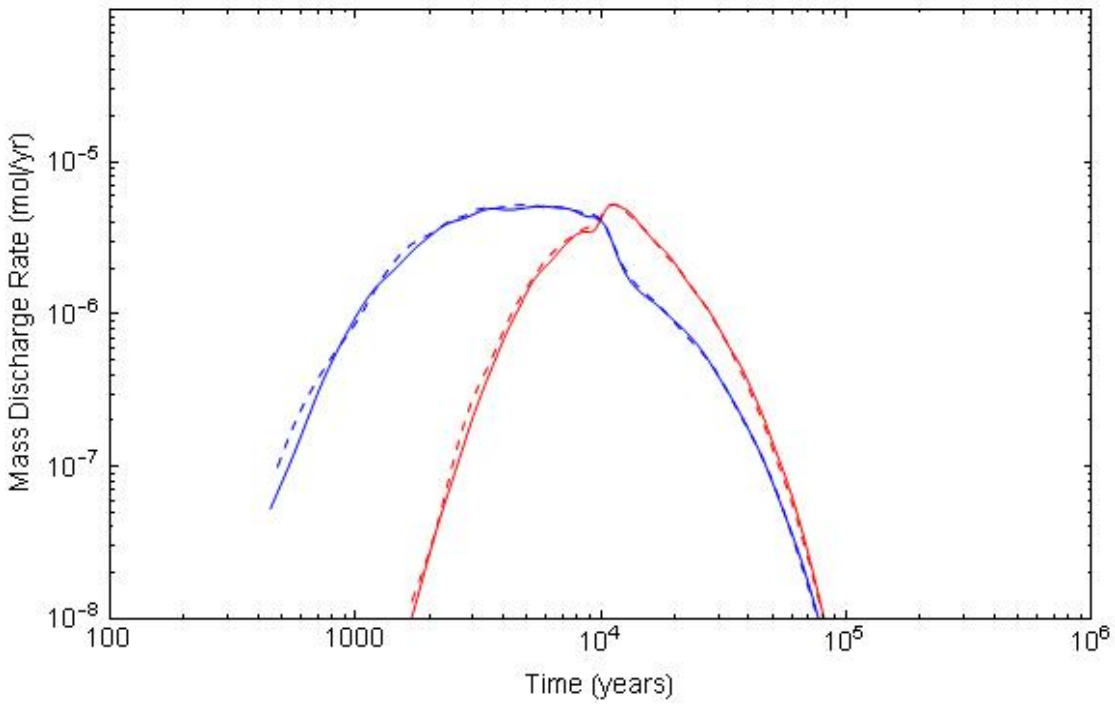

Figure 13. *Result of Version 3.3.1 Verification Test 2. The solid lines show the results of MARFA 3.2.3, and the dashed lines show the results of MARFA 3.3.1. The red lines are for radionuclide A, and the blue lines are for radionuclide B. The time of the flow change is at 10,000 years.*

6.3 Multiple Sources and Multiple Pathways Test

This test was designed to test the capability of MARFA 3.3.1 to properly evaluate multiple sources and multiple pathways. The test evaluates a release from two separate sources, with releases starting at differing times and onto different pathways. The first source and pathway are identical to the source and pathway described in Test1a. The second source releases a 1-mol pulse release of radionuclide *B* at 10,000 years onto a second pathway that has a transport length of 50 m and a transport resistance parameter of 5×10^5 yr/m. The radionuclide-specific retention model and parameters are the same as those for Test1a.

The results of this test are shown in Figure 14. These results show good agreement between MARFA 3.3.1 and MARFA 3.2.3.

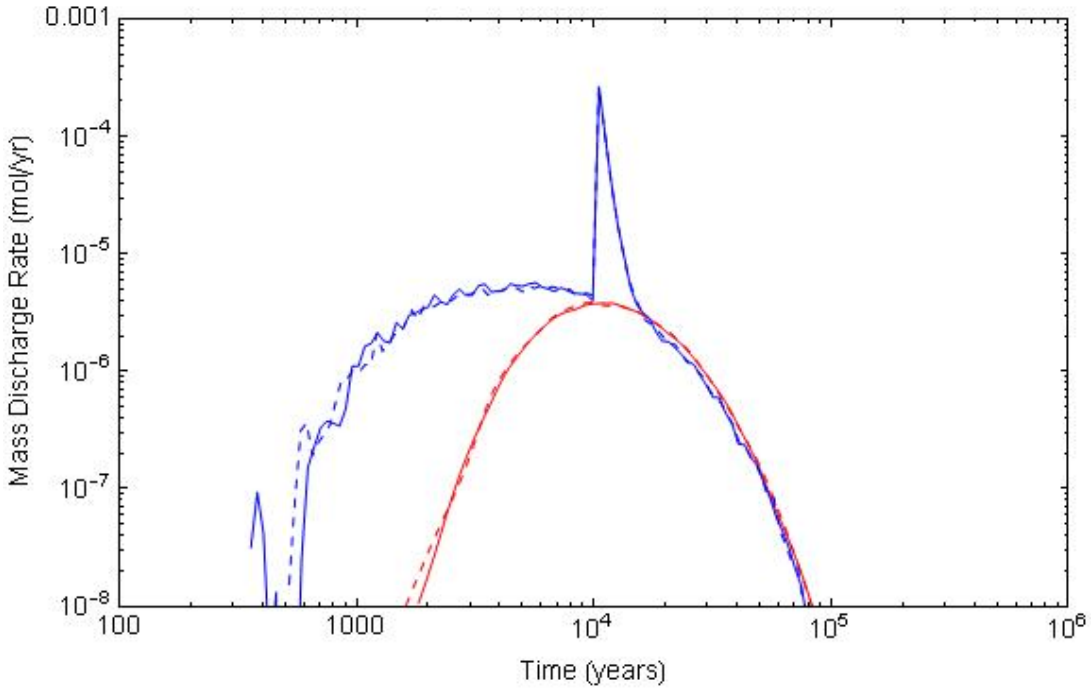

Figure 14. *Result of Version 3.3.1 Verification Test 3. The solid lines show the results of MARFA 3.2.3, and the dashed lines show the results of MARFA 3.3.1. The red lines are for radionuclide A, and the blue lines are for radionuclide B. The statistical noise in the early breakthrough of radionuclide B is typical of Monte Carlo analyses. The noise is amplified in this plot because the breakthrough reconstruction algorithm assigned a small bandwidth to resolve the peak at 10^4 years, thereby increasing the sensitivity to statistical noise in the remaining part of the curve.*

7 EXAMPLE SIMULATIONS

MARFA Version 3.2.3 includes complete input for an example simulation that is designed to activate the entire range of capabilities. This example uses 120 pathway trajectories from a nested DFN/CPM simulation provided by Serco Assurance [Joyce, 2006]. The trajectories are shown in Figure 15. Note some of the trajectories originate outside the DFN region that surrounds the hypothetical repository. These isolated trajectories are the result of an error in specifying the trajectory starting locations within the CONNECTFLOW simulation [Joyce 2006]. In the example, the VALID TRAJECTORIES option (see Appendix A) is used to force MARFA to skip these spurious trajectories. There are 91 valid trajectories. The valid trajectories are replotted in Figure 16.

The simulation includes 13 rock types. One of these, representing the explicit DFN region, is deterministic. The other rock types are stochastic. Subgrid trajectories are included with the input set. Limited diffusion is used as the retention model in all rock types.

The example has a glacial period from 10,000 to 11,200 years. The first and last 100 years of the glacial period are high-flow flushing periods wherein the flow velocity increases by a factor of 50 relative to the base flow. In the intervening 1,000 yr period (glacial nonflushing period), the flow is reduced to 10% of the original value.

The decay chain for this example is $^{241}\mathrm{Am} \rightarrow ^{237}\mathrm{Np} \rightarrow ^{233}\mathrm{U} \rightarrow ^{229}\mathrm{Th}$. Radionuclides are introduced as ^{241}Am in a short pulse at t=0. The total mass released is 1 mol.

Input files for the example are included with the MARFA 3.2.3 code in the directory AcceptanceTests/Test1. This example may serve as an installation test.

A variant of this example is also provided. Acceptance Test 1a is identical to Test 1 except that the flow field is steady (i.e., no glaciation modeling case). Breakthrough of ^{237}Np as calculated by MARFA for these two modeling cases is compared in Figure 17. Spikes corresponding to the flushing periods at 10,000 and 11,100 years are apparent in the breakthrough for the glaciation modeling case, but the total mass released in these spikes is relatively small compared to the main breakthrough around 10^6 years.

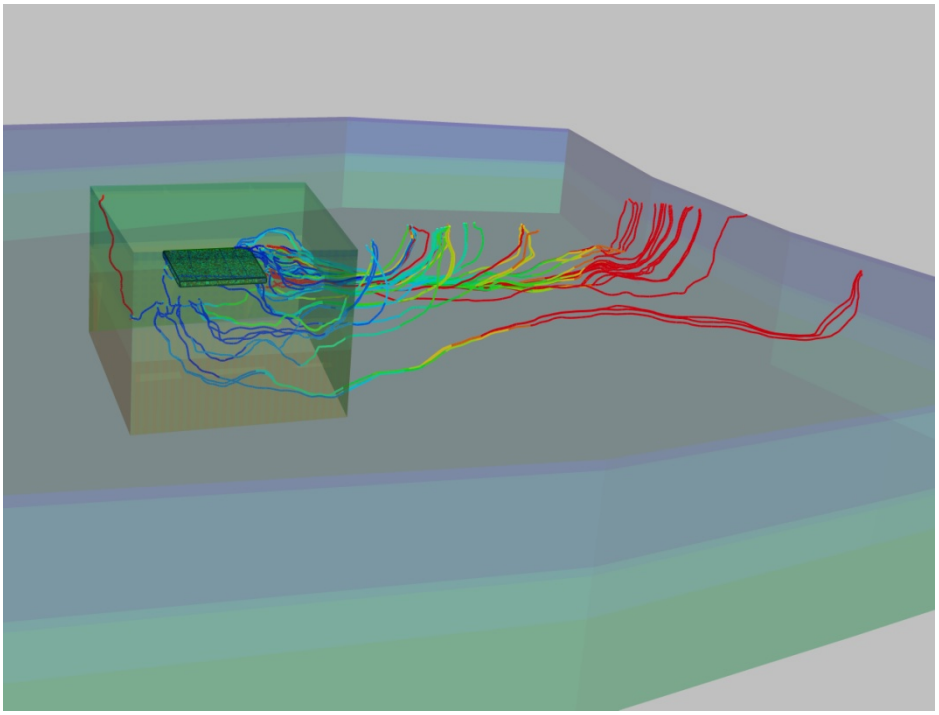

Figure 15. *CONNECTFLOW trajectories forming the pathways for the MARFA example simulation. Trajectories are color coded by cumulative groundwater travel time. This graphic was provided by Serco Assurance [Joyce 2006].*

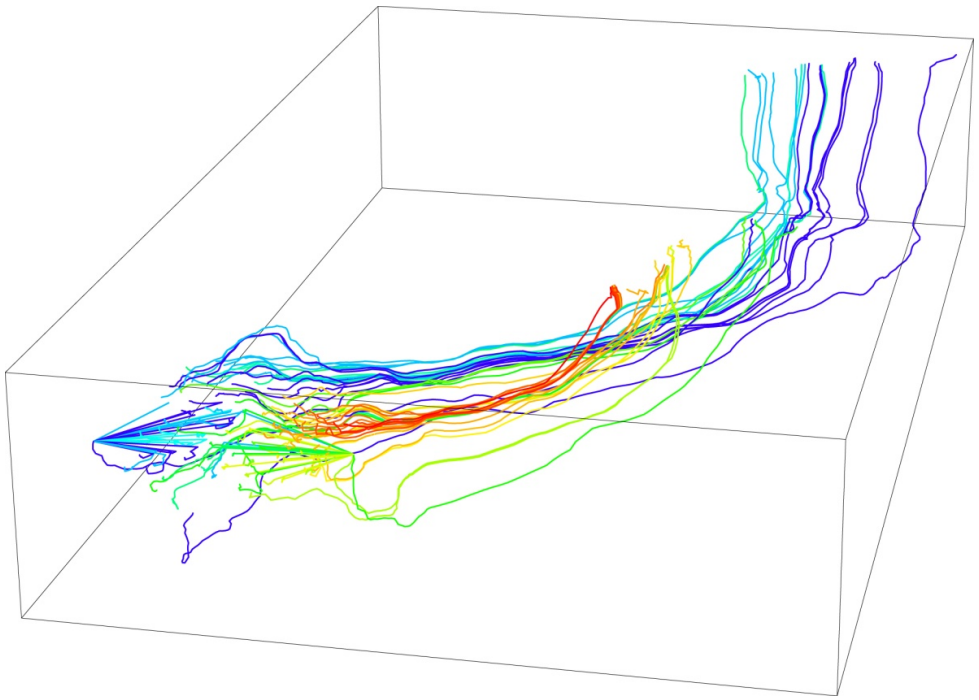

Figure 16. *Valid pathways used in the example simulation color coded by cumulative radionuclide discharge as calculated by MARFA. Warmer colors denote higher discharge. A small number of pathways are responsible for the majority of the flux in this example.*

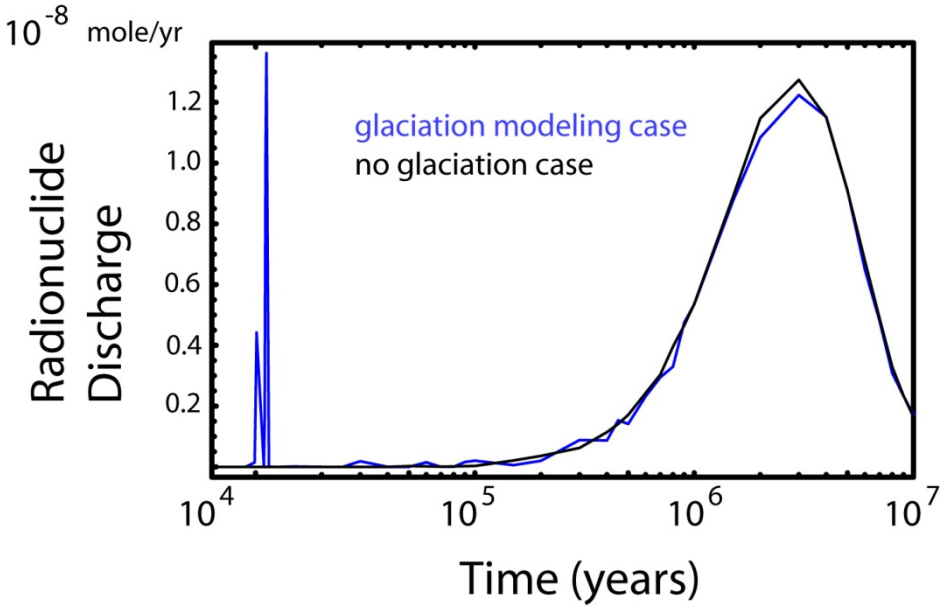

Figure 17*. Radionuclide discharge for ^{237}Np for example simulations with and without flow field changes due to glacier passage.*

Results of a second example simulation demonstrating the particle splitting capabilities in MARFA 3.2.3 are shown in Figure 18. This example uses the decay chain $^{238}\text{U}\rightarrow^{234}\text{U}\rightarrow^{230}\text{Th}\rightarrow^{226}\text{Ra}\rightarrow^{210}\text{Pb}$. The transport pathway length is 500 m, groundwater velocity is 12.5 m/yr, half aperture is 1 mm, matrix penetration depth is 3 cm, matrix porosity is 0.001, Peclet number is 25, and effective diffusion coefficient is 6.0×10^{-7} m^2/yr. The matrix retardation factor is 2×10^6 for U, Ra and Pb and 3×10^6 for Th. The radionuclide source into the pathway is of strength $0.001\exp(-t/1000)$ mol/yr for ^{238}U, where t is in years. The source strength is zero for the other radionuclides. It is important to note that this is a difficult transport simulation for particle-based methods because the very long half-life of ^{238}U means that a very tiny fraction of the initial mass decays en route, making it difficult to get statistically significant Monte Carlo estimates of the radionuclides further down the decay chain. The particle splitting algorithm in MARFA was used, splitting 1000 to 1 for every ^{238}U decay event and 100 to 1 for all other decay events. A total of 10^8 particles was used. Results in Figure 18 compare favorably with independent numerical solutions (see Figure 38 of Robinson and Watson [2011]). This good agreement demonstrates that reliable estimates of breakthrough can be obtained even with difficult decay chains. It is important to note that this demanding example is contrived and highly unlikely to be encountered in practice because of ingrowth radionuclide decay in the near field, which would cause the source of progeny radionuclides to be non-zero.

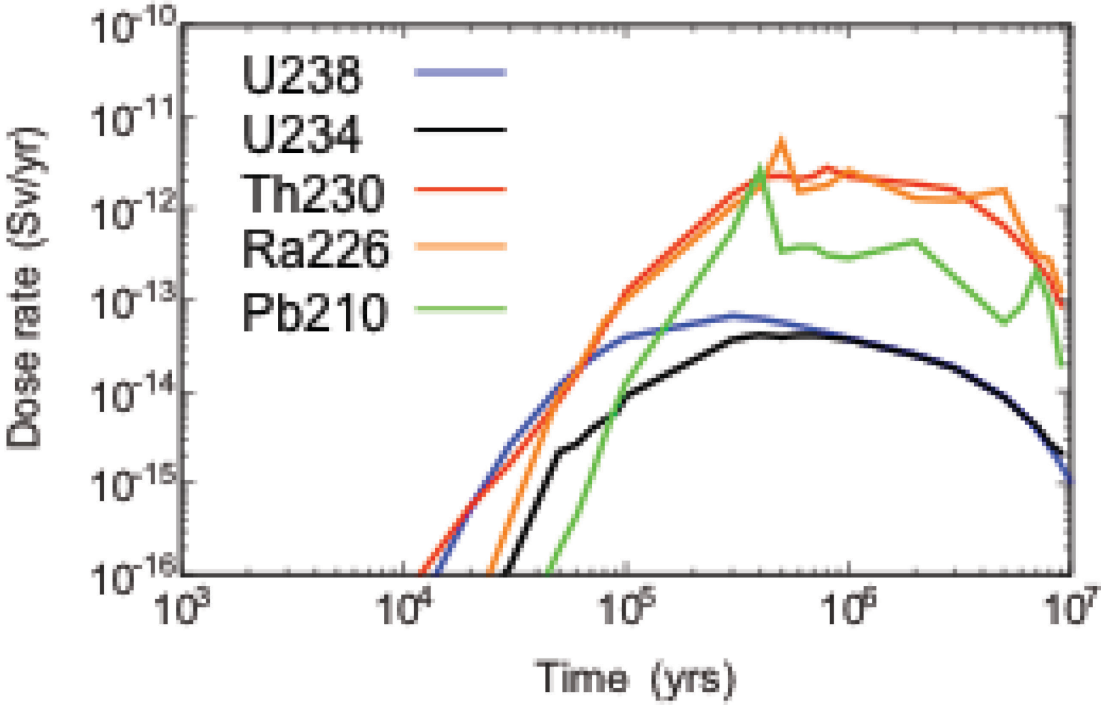

Figure 18. *Radionuclide discharge for an example demonstrating the use of particle splitting. Very minor mass of ^{238}U is generated on the transport pathway, making it difficult to get reliable estimates of breakthrough for progeny radionuclides. Particle splitting allows reliable estimates to be obtained.*

One realization of the Forsmark fracture system was used to evaluate potential effects of flow direction changes on transport and to test MARFA 3.3.1 in a realistic configuration. A 100 m × 100 m × 100 m block was used (Figures 19 and 20). Node-routing data (PTH files) for two flow configurations were used, corresponding to flow in the *x* and *y* directions. The radionuclides ^{135}Cs, ^{129}I, and ^{59}Ni and the unlimited matrix diffusion model were used. These three nuclides span the relevant ranges of decay constants and retardation parameters $\kappa = \sqrt{\theta_m R_m D_{eff}}$, where D_{eff} [m^2/yr] is the matrix effective diffusion coefficient, R_m [-] is the matrix retardation factor, and θ_m [-] is matrix porosity (Table 1). The source strength for each radionuclide is 1 mol/yr starting at 1,000 years decreasing promptly to 0 mol/yr after 1,010 years. Thus, a total of 10 moles of each radionuclide is released into the nodes that reside on the surface of the release face.

The applied head gradient was initially in the *x* direction; particles were released on the upstream face. The magnitude of the applied head gradient was 0.0001 m/m. The applied gradient direction was changed to the *y* direction at 10^4 years. Input files for the example are included with the MARFA 3.3.1 code in the directory AcceptanceTests/Test1.

Table 1. *Radionuclides and Corresponding* κ *Parameters and Half-Lives in the Forsmark 100-m Block Simulations.*

Radionuclide	κ [m $yr^{-1/2}$]	**Half-Life** (years)
Cs-135	7.10×10^{-3}	2.3×10^{6}
I-129	2.10×10^{-5}	1.57×10^{7}
Ni-59	3.48×10^{-3}	7.6×10^{4}

Figure 21 shows the mass discharged from the block when particles were released on the entire upstream face. These figure show that iodine responds to direction changes with prompt decreases in discharge rates, while the cesium and nickel respond with prompt increases in discharge rates. This behavior is attributed to the differing mass distributions within the discrete fracture network. For the mobile iodine, the majority of the mass has already discharged before the flow changes occur; the remaining plume is skewed toward the downstream face. After the flow direction change, the average distance to the new discharge face promptly increases, causing a prompt decrease in the discharge rate. For the relatively immobile cesium and nickel, the opposite occurs. Specifically, some fraction of the plume is located near the edge of the domain at the time of the flow change and thus has only a short distance to travel to the new downstream face after the flow change. This effect causes the abrupt increase in discharge for cesium and nickel.

For this simulation, 5 million particles were launched. Approximately 2.8 million particles reached a discharge boundary. The maximum number of visits to any given node was set to 3. Of the 5 million particles, approximately 4 % were terminated due to excessive visits to a node (nonphysical looping caused by inaccuracy in the input flow field). Of the surviving particles, approximately 40 % passed through at least one nonphysical loop. However, the total amount of time spent in loops was relatively small (approximately 0.1 %) compared with the total residence time in the network. That the total amount of time spent in loops and the total number of particles experiencing excessive loops are small is an indication that artifacts introduced by loops in the flow field are not significant.

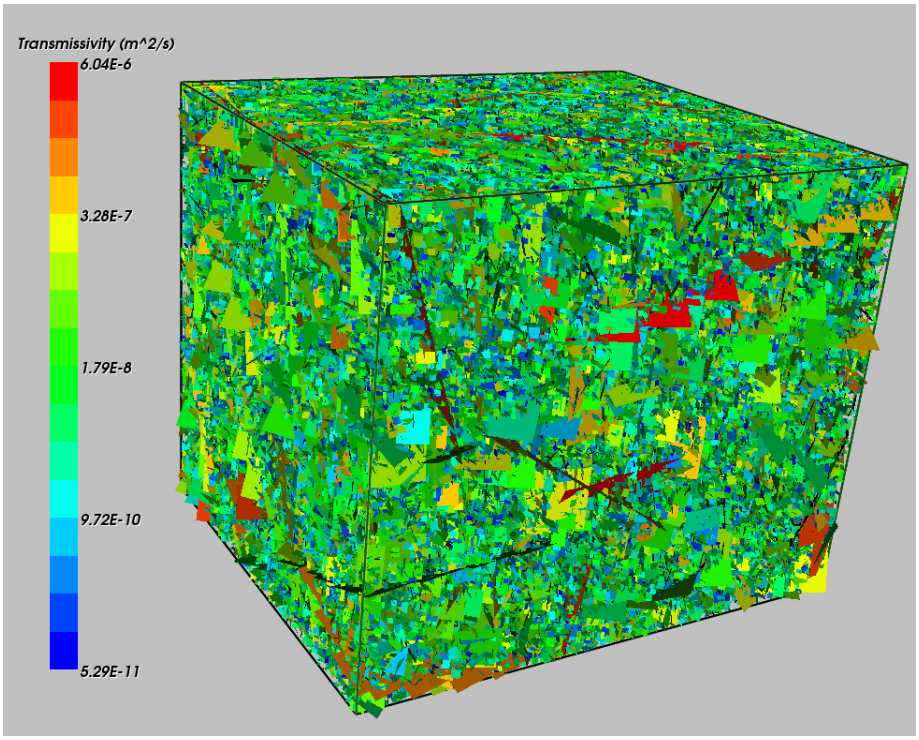

Figure 19. *Fracture Network for the Forsmark 100-m Block Simulation.*

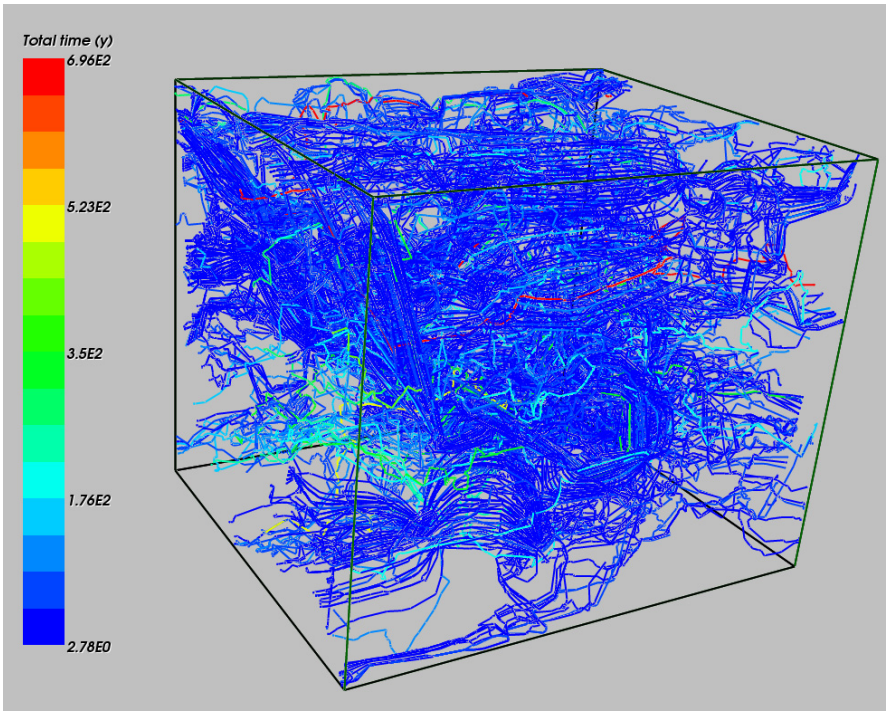

Figure 20. *Transport Pathways as calculated by CONNECTFLOW for the Forsmark 100-m Block Simulation. Flow is from left to right.*

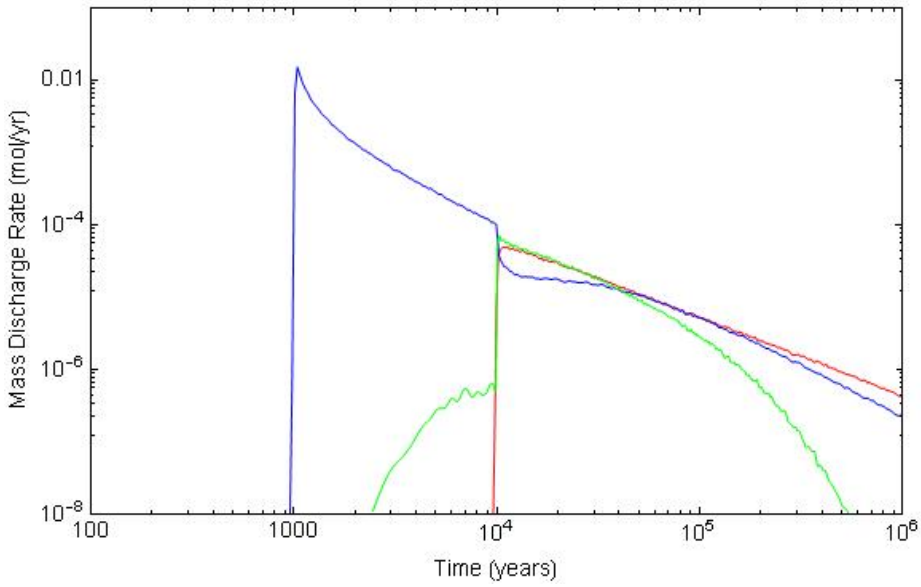

Figure 21. *Result of the Forsmark 100-m Block Simulation. The red, blue, and green lines represent ^{135}Cs, ^{129}I, and ^{59}Ni discharge summed over all boundaries. The radionuclide source begins releasing at 1,000 years, with a flow direction change (x direction to the y direction) at 10,000 years.*

8 SUMMARY

This document describes input requirements for MARFA Versions 3.2.3 and 3.3.1. Version 3.2.3 simulates transport of radionuclides along pathways originating from multiple source locations. Velocities and sorption properties of the pathways are piecewise constant in time. There are no restrictions on the number of radionuclides, the length or number of the decay chains, the number of trajectories, or the number of flow changes. Branching chains are not supported. Full heterogeneity in the flow-related properties of the pathway is supported. In addition, an optional stochastic downscaling mode is available. The downscaling algorithm is designed to recover the neglected effects of subgrid scale velocity fluctuations when the pathway flow properties have been determined from an upscaled flow model. Heterogeneity in the chemical retention properties is addressed by rock type. That is, the user must define a set of rock types with constant retention properties in each. There is no limit on the number of rock types. The equilibrium sorption, limited diffusion, and unlimited diffusion models are supported.

The primary limitation of MARFA Version 3.2.3 is that it cannot accommodate trajectories that change in time. That is, the velocity may change in magnitude, but not in direction. Approaches for fully transient flow fields are incorporated in Version 3.3.1.

REFERENCES

Carslaw, H.S. and J.C. Jaeger, "Conduction of Heat in Solids, 2nd Edition", Oxford University Press, New York, 1959.

Crawford, J., I. Neretnieks and M. Malmström, "Data and Uncertainty Assessment for Radionuclide Kd Partitioning Coefficients in Granitic Rock for Use in SR-Can Calculations," SKB Report R-06-75, Stockholm, Sweden: SKB, 2006.

Cvetkovic, V., G. Dagan, and H. Cheng, Contaminant Transport in Aquifers with Spatially Variable Flow and Sorption Properties, *Proc. R. Soc. Lond. A*, *454*, 2173–2207, 1998.

Cvetkovic, V., J.-O. Selroos, and H. Cheng, Transport of Reactive Tracers in Rock Fractures, *J. Fluid Mech., 378*, 335–356, 1999.

Cvetkovic, V., and R. Haggerty, Transport with Multiple-Rate Exchange in Disordered Media, *Phys. Rev. E 65*(5), 051308–051317, 2002.

Dagan, G., and V. Cvetkovic, Reactive Transport and Immiscible Flow in Geological Media 1. General Theory, *Proc. R. Soc. Lond. A*, *452*, 285–301, 1996.

Hartley, L.J. and D. Holton, "CONNECTFLOW (Release 2) Technical Summary Document," Oxon, Oxford, England: Serco Assurance, 2003.

Hartley, L.J., I. Cox, D. Holton, F. Hunter, S. Joyce, B. Gylling, and M. Lindgren, "Groundwater Flow and Radionuclide Transport Modelling Using CONNECTFLOW in Support of the SR Can Assessment," SKB Report R-04-61, Stockholm, Sweden: SKB, 2004.

ISO/IEC, "Fortran Enhanced Data Type Facilities," ISO/IEC Technical Report 15581, International Standards Organization, 1998.

Jacquet, O. and P. Siegel, "Regional Groundwater Flow Model for a Glaciation Scenario. Simpevarp Subarea - Version 1.2," SKB Report R-06-100, Stockholm, Sweden: SKB, 2006.

Joyce, S., Private Communication, September 9, 2006.

Painter, S., V. Cvetkovic, and O. Pensado, "Time-Domain Random Walk Method for Simulating Radionuclide Transport in Fractured Porous Rock," Proceedings of the 2006 International High-Level Radioactive Waste Management Conference, Las Vegas, Nevada, April 30–May 4, La Grange Park, Illinois: American Nuclear Society, 2006.

Painter, S., J. Mancillas, V. Cvetkovic, and O. Pensado, "Time-Domain Particle Tracking Method for Simulating Transport with Retention and First-Order Transformation," *Water Resources Research*, 44, doi:10.1029/2007WR005944, 2008.

Painter, S. “Accommodating Transient Velocities in Time-Domain Particle Tracking Algorithms of Radionuclide Transport,” Proceedings of the 12th International High-Level Radioactive Waste Management Conference (IHLRWM); 7–11 September 2008; Las Vegas, Nevada. LaGrange Park, IL: American Nuclear Society (CD ROM). 2008.

Painter, S. and V. Cvetkovic, “Upscaling Discrete Fracture Network Simulations: An Alternative to Continuum Transport Models,” *Water Resources Research 41*, doi:10.1029/2004WR003682, 2005.

Robinson, P. and C. Watson, “Handling Interfaces and Time-varying Properties in Radionuclide Transport Models”, SSM Report Reseach 2011:11, Swedish Radiation Safety Authority, 2011.

Sudicky, E.A. and E.O. Frind, “Contaminant Transport in Fractured Porous Media: Analytical Solution for a Two-Member Decay Chain in a Single Fracture,” *Water Resources Research* 20(7), 1021–1029, 1984.

Silverman, B.W., “Density Estimation for Statistics and Data Analysis,” Chapman and Hall, New York, 1986.

APPENDIX A: MARFA 3.2.3 INPUT

The units for all pathway and radionuclide property input parameters are various combinations of meters and years. The radionuclide source rate should be specified in mol/yr or Bq/yr.

Conventions used in this Appendix are as follows. Input parameter names are shown in **bold** font. File names are shown in *italics*. The `courier` font is used when specifying input file formats. A line consisting of a single colon in the input block means that lines are skipped. An ellipsis (…) indicates that the item is to be repeated. Any text following an exclamation point in an input block is to be regarded as a comment or explanation.

Four input files are required for all MARFA runs. Users must specify radionuclide decay chains in the *nuclides.dat* file, the set of rock types in the *rocktypes.dat* file, radionuclide sources in *sources.dat*, and the pathway properties in *trajectories.dat*. These four files must exist and must reside in the run directory.

If the downscaling algorithm is to be used (i.e., at least one rock type is specified as stochastic), then a set of subgrid trajectories must also be provided. In contrast to the other input files, the subgrid trajectories data need not reside in the run directory. The location (path) of the subgrid trajectories directory is specified in the *rocktypes.dat* file.

In addition to the user-supplied input, MARFA also requires access to a set of files that contain data for lookup tables used in sampling the retention times. The required data files are included as part of MARFA Version 3.2. The location (path) of the required data files on the local file system must be specified in the *trajectories.dat* file.

Any input file may start with an optional header section. The header section contains an arbitrary number of header lines. A header line is identified by an exclamation point "!" as the first non-black character. Comment lines are not permitted after the input starts. Partial lines of comment may follow the required data fields on a line, however. The following subsections give the required data format after the header section.

A.1 Specifying the Decay Chains

The format for the *nuclides.dat* file is

```
nelem
elem_name !one line for each element(nelem times)
:
nnuc
nuc_name  lambda  associated_elem  nextnuc  imprtnce  nsplit
!above line repeated nnuc times
```

Here **nelem** is the number of elements in the decay chain, and **elem_name** is the name of the element. The list of element names is used only to associate each nuclide with the appropriate element. Retention properties are entered for each element, as described in the input for the *rocktypes.dat* file.

The input **nnuc** is the number of nuclides to be included in the transport calculation. Each nuclide is assumed to have, at most, one daughter nuclide (branching in a decay chain is not supported). The inputs **nuc_name** and **lambda** are the nuclide name and decay constant [T^{-1}] for the nuclide, and **associated_elem** is the name of the associated element. The next nuclide in the decay chain is specified by the **nextnuc** parameter. The value NULL for a **nextnuc** parameter indicates that the nuclide is at the end of the chain. Each specified **associated_elem** must be in the list of element names, and each **nextnuc** must be in the list of nuclide names.

The input **imprtnce** is a user-specified relative importance factor that may be used to force some nuclides to be sampled more frequently at the source. This option is useful, for example, for obtaining higher accuracy for risk significant nuclides. The probability of selecting a given nuclide is proportional to the total source strength for that particle multiplied by its **imprtnce** value. MARFA assigns each particle of a given nuclide type a statistical weight inversely proportional to the nuclide's **imprtnce** value, thus ensuring that the resulting estimate is unbiased. As with any importance-weighted Monte Carlo sampling scheme, an informed choice for the importance value may decrease the variance in the result, but a poor choice will increase the variance.

The input **nsplit** is an optional parameter controlling splitting at each decay event. If it is present and greater than 0, each particle will be split into nsplit+1 particles upon a decay event. This option is useful when modeling transport of decay chains when one of the radionuclides has a very long half-life.

Note the nuclide list may include multiple chains and nuclides that do not belong to any chain. The user simply needs to specify the decay product for each radionuclide. Converging chains are supported, but branching decay chains are not.

An example *nuclides.dat* file is shown in Figure A-1. This example has 7 radionuclides and two chains: $^{241}Pu \rightarrow {}^{241}Am \rightarrow {}^{237}Np \rightarrow {}^{233}U$ and $^{246}Cm \rightarrow {}^{238}U$. The radionuclide ^{99}Tc is not modeled as part of a chain.

```
6
Pu
Am
Np
Cm
U
Tc
7
Pu241          4.8e-2       Pu   Am241    1.0
Am241    1.6e-3       Am    Np237     1.0
Np237          3.2e-7       Np   U233       1.0
U233           4.37e-6       U          NULL      1.0
Cm246    1.46e-4     Cm    U238       1.0
U238     1.55e-10     U    NULL      1.0
Tc99     3.25e-6      Tc   NULL      1.0
```

A.2 Specifying the Radionuclide Source

The radionuclide source is specified in the *source.dat* file. The source is to be specified as a source rate in mol/yr or Bq/yr. The format is

```
npart
units                ! mol/yr or Bq/yr
samplemethod         ! keyword controlling source sampling
nsources             ! number of sources
sourceID             ! beginning of source block
traj1 traj2          ! Range of trajectories associated with
                     source
ndpts
time RN1 RN2 RN3 …   ! one for each nuclide
:                    !  repeat  above  line  for  a  total  of
ndpts lines
:                    ! repeat above source block for a total
                     of nsources blocks
```

The parameter **npart** is the total number of particles to be used. The **units** parameter is used to specify the source units. The options are “mol/yr” or “Bq/yr”. The optional keyword **samplemethod** controls the sampling of the source. The default method is to sample the source history with uniform statistical weights for each particle of a given radionuclide. If the **samplemethod** keyword is present and specified as “UNIFORM IN TIME”, sampling of the source will be uniform in time with appropriate statistical

weighting applied to each particle, as described in Section 3.3 of this document. The parameter **nsources** is the number of source locations (e.g., failed canisters). For each source the **sourceID** is a 10-character identifier for that source. The parameters **traj1** and **traj2** identify the upper and lower indices in the trajectory list for those trajectories that will be associated with each source. The number of time points in a source history is **ndpts**. For each value of **time**, **nnuc** values of the source release rate are read, where **nnuc** is the number of nuclides (see Section A.1).

An example *source.dat* file is shown in Figure A-2. This example has two source locations and a dirac-δ function input, which is approximated as a square pulse of duration 0.1 years. The source strength is 0.001 mol/yr for the first radionuclide and 0.005 mol/yr for the second nuclide in the two-member chain, where the net cumulative source strength is computed using a linear interpolation of the defined mass release rates.

```
100000      !  Number of particles
mol/yr        !  Units
2               !  Two source locations
Can1          !  Canister 1
1  20          !  Canister 1 is associated with trajectories 1 through
20
2               !  Two times are used to specify source
0.0  0.001 0.005  !  Source strength for RN1 and RN2  at 0 years
0.1  0.001 0.005  !  Cumulative source for RN1 and RN2 at 0.1 years
Can2          !  Canister 2
21-40         !  Canister 2 is associated with trajectories 21-40
2                          !  Number of times used to specify source
0.0  0.001 0.005  !  Source strength for RN1 and RN2 at 0 years
0.10.001 0.005  !  Cumulative source for RN1 and RN2 at 0.1 years
```

A.3 Defining Rock Types

The *rocktypes.dat* file specifies the retention model and retention properties for each rock type. In addition, the file specifies stochastic or deterministic mode for each. The format is

```
numtypes          ! number of rock types defined
datadir           ! path to data directory for subgrid
                  trajectory files
typeID            ! length 5 character string identifier
                  for the rock type
ret_model
ret_parameters ! retention parameters for element 1
```

```
:                ! above line is repeated for each element
:                ! repeat block for each flow period
                 alpha stochasticflag numdir numvc
:
:                ! repeat block for a total of numtypes
```

Here **numtypes** is the number of rock types modeled. The **datadir** parameter provides the file system path to the subgrid trajectories data files for stochastic rock types. The **datadir** parameter is read but not used if all rock types are deterministic. The input parameter **typeID** is a 5-character string identifying the rock type. The input block headed by a **typeID** is repeated for a total of **numtype**. If retention parameters are modeled as constant in the entire domain, then only one rock type needs to be defined.

The retention model is specified by the **ret_model** parameter. Allowed values are MD (unlimited matrix diffusion), LD (limited matrix diffusion), ES (equilibrium sorption), and TAB (tabular). The next **nelem** lines contain the retention parameter values at the start of the simulation. Retention parameters for the ES, MD and LD options are read in the same order as the element list in *nuclides.dat,* with one set of retention parameters for an element on each line. The input read from these lines depend on which retention model is specified. If ES is specified, then the dimensionless retardation factor *R* is read for each element. Here $R = 1 + \frac{\rho_b K_d}{\theta}$ where ρ_b, K_d and θ are bulk density [M/L^3], equilibrium distribution coefficient [L^3/M] and porosity [-] for the porous medium, respectively. If MD is specified, then κ is read for each element where $\kappa = \sqrt{\theta_m D_{eff} R_m}$ [L T$^{-1/2}$], θ_m is the matrix porosity [-], D_{eff} is the matrix effective diffusion coefficient [L^2/T], and R_m is the matrix retardation factor [-]. If LD is specified, then the three parameters κ, η and k_a are read from each line. Here $\eta = \frac{\Delta}{D_{eff}}$ [T/L] and Δ [L] is the size of the matrix region accessible by diffusion, and k_a [1/L] is a sorption coefficient for equilibrium sorption on fracture surfaces.

Starting with Version 3.2.3, a tabular retention model is available (TAB) option. This option allows the user to specify a retention time distribution for each element as a lookup table. The format is different from the other retention models in that the TAB option redirects input to a user-specified directory. Details of this option are provided in Appendix A-4.

MARFA Version 3.2 allows velocity and rock properties to be piecewise constant in time. If this option is invoked (see Section A.4), then retention properties must be entered for each flow period. The retention parameters should be entered for each flow period, one flow period per line. The number entered must be consistent with the *trajectories.dat* file.

The parameter **alpha** is longitudinal dispersivity [L]. If **alpha** is entered as "0," the longitudinal dispersion calculation is skipped. Note that the longitudinal dispersion calculation may increase the run time significantly.

The single-character input field **stochasticflag** indicates whether the rock type is stochastic or deterministic. Allowed values are S for stochastic or D for deterministic. For deterministic rock types, no other information is read. For stochastic rock types, the **numdir** and **numv** parameters are read from the same line. The **numdir** parameter specifies the number of directions for the subgrid trajectories (see Section 2). A separate data file with subgrid trajectories is required for each of the **numdir** directions, as explained in Section A.5. The **numv** parameter is a numerical parameter used in the stochastic downscaling algorithms. Details of the **numv** parameter are provided by Painter and Cvetkovic [2005].

Example *rocktypes.dat* files are shown in Figures A-3 and A-4. Three rock types are defined in this example: TUN1, DFN1, and CPM1. The TUN1 rock type uses the equilibrium sorption model, has a dispersivity of 2 m, and is deterministic. A rock type defined this way might represent, for example, a backfilled repository tunnel. The DFN1 rock type is deterministic and uses the limited diffusion model (e.g., an explicitly represented fracture network). The CPM1 rock type also uses the limited diffusion retention model, but is stochastic (e.g., a continuous porous medium region). The downscaling algorithm is used for this rock type. The subgrid trajectories for the downscaling algorithm is read from the directory */home/spainter/MARFA31/SubgridTrajectories/CPM1/*. One set of subgrid trajectories will be used (**numdir** = 1), which presumes an isotropic network. A total of eight velocity groups (**numvc** = 8) is used in the downscaling algorithm.

```
3            !  Number of rock types
/home/spainter/MARFA31/SubgridTrajectories/
TUN1               !  Rock type name
ES                  !  Retention model (ES/LD/MD)
500   ! retention parameters for element 1
100   ! retention parameters for element 2
200
200
2. D               !  Dispersivity, stoch/det flag (S/D)
DFN1
LD
 0.006 2.0e5 0.0
 0.008 2.0e5 0.0
                          0.004 2.0e5 0.0
 0.003 2.0e5 0.0
 0. D
CPM1
LD
 0.006 2.0e5 0.0
 0.008 2.0e5 0.0
 0.004 2.0e5 0.0
 0.003 2.0e5 0.0
 0. S 1 8
```

```
3            !  Number of rock types
/home/spainter/MARFA31/SubgridTrajectories/
TUN1               !  Rock type name
ES                  !  Retention model (ES/LD/MD)
500  ! retention parameters for element 1, flow period 1
100
200
200
500  ! retention parameters for element 1, flow period 2
100
200
200
2. D               !  Dispersivity, stoch/det flag (S/D)
DFN1
LD
 0.006 2.0e5 0.0 ! retention parameters for element 1, flow period 1
 0.008 2.0e5 0.0
 0.004 2.0e5 0.0
 0.003 2.0e5 0.0
 0.006 2.0e5 0.0 ! retention parameters for element 1, flow period 2
 0.008 2.0e5 0.0
 0.004 2.0e5 0.0
 0.003 2.0e5 0.0
 0. D
CPM1
LD
 0.006 2.0e5 0.0 ! retention parameters for element 1, flow period 1
 0.008 2.0e5 0.0
 0.004 2.0e5 0.0
 0.003 2.0e5 0.0
 0.006 2.0e5 0.0 ! retention parameters for element 1, flow period 2
 0.008 2.0e5 0.0
 0.004 2.0e5 0.0
 0.003 2.0e5 0.0
 0. S 1 8
```

Figure A-4. *Example rocktypes.dat file with two flow periods (one flow change) and 4 elements.*

A.4 Defining Rock Types using the TABular option

Starting with Version 3.2.3, a tabular retention model is available (TAB) option. This option allows the user to specify a retention time distribution for each element as a lookup table. The format for the *rocktypes.dat* file with the tabular option is

```
numtypes        ! number of rock types defined
datadir         ! path to data directory for subgrid
                trajectory files
typeID          ! length 5 character string identifier for
                the rock type
TAB
Rtdirname1      ! directory with retention distributions for
                flow period 1
:               ! above line is repeated for each flow
                period
alpha stochasticflag numdir numvc
:
:               ! repeat block for a total of numtypes
```

The format is the same as described in Section A.3 except for the TAB keyword and the next few lines. The TAB keyword is literal. Following that are **nchanges**+1 lines, where **nchanges** is the number of flow changes (see next Section A.5). For each flow period, the filepath **rtdirname** to a directory containing the retention time distribution lookup tables must be specified.

Each specified directory should contain one file for each element. The file will be named **retdist.dat1** for element 1, **retdist.dat2** for element 2, etc. Each of these files contains the retention time distribution conditional on β. The format is

```
numbeta         ! number of beta values for which tabular
                values are stored
beta            ! beta value
nquantiles      ! number of quantiles used to define the
                distribution
Q1 T1           !quantile, retention time
:               !repeat above line for a each quantile-time
                combination
:               !repeat above blocks for a total of numbeta
                vslues
```

In addition, the directory should contain one file named *auxdata.dat.* This format for this file is

```
Delta           ! size of the matrix accessible region (m)
                ka Dbar  !
:               ! repeat above line for each element
```

In the *auxdata.dat* file, the parameter **Delta** is the size of the accessible matrix region. For a multilayer matrix, this is the sum of all matrix thicknesses. The parameter **ka** is k_a [1/L], a sorption coefficient for equilibrium sorption on fracture surfaces. **Dbar** is a weighted diffusion coefficient [m^2/yr] across the matrix layers. **Dbar** and **Delta** are used only when there are flow changes.

Although a user may specify any retention time distribution through the tabular option, the primary intended use is for multilayer matrix diffusion models. Included with Version 3.2.3 is a helper application in the form of two Mathematica scripts *ThreeLayerRetention.m* and *3LayerNoInterface.m*. These scripts would typically be called by a third user-customized script to create the retention time distributions for a 3-layer matrix model. The helper applications are currently configured to produce tables ranging from β=1 to 10^7 yr/m. Included with Verification Tests 9a is an example of such a script that can serve as a template for using the helper application.

A.5 The Trajectories.dat File

Trajectories, as calculated from CONNECTFLOW nested DFN/CPM regional scale models, are read from a file named *trajectories.dat*. Two possible formats of this file are available. With the first format, the trajectory data are fully contained within the *trajectories.dat* file. With the second format, the *trajectories.dat* file contains the location of a CONNECTFLOW data file containing the trajectory information.

Format 1 is described as follows.

```
retdatadir
ntraj                       ! number of trajectories
TrajID  xloc yloc zloc      ! Trajectory ID and trajectory
                            location
xe ye ze typeID tau beta cpmgradh
          ! segment data
:         ! repeat above line for each segment
END       ! end of trajectory delimiter
:         ! above block repeated for each trajectory
```

The **retdatadir** parameter provides the file system path to the generic lookup-table data required for sampling the retention time distributions. The parameter **ntraj** is the number of trajectories; **TrajID** is a 10-character label identifying the trajectory; and **xloc**, **yloc**, and **zloc** are the x-, y-, and z-positions for the trajectory head. The **TrajID** is employed by MARFA to identify specific trajectories, such that each **TrajID** needs to be unique. Sources are associated to trajectories by their sequential order, matched to the trajectory range defined in *source.dat*. The data read for each segment include the location of the end of the segment (**xe,ye,ze**) and a 5-character identifier for the rock type (**typeID**).

The advective travel time (**tau**) and the F-quotient (**beta**) are also read. If the rock type is a stochastic one, then **tau** and **beta** represent CPM (upscaled) values; otherwise, they are DFN values for the segment. If the particular rock type is a stochastic one, then the magnitude of the hydraulic gradient (**cpmgradh**) at that location is also read. If the particular rock type is a deterministic one, then **cpmgradh** is not used. The hydraulic gradient is used to adjust (scale) the stochastically generated segment properties to the local macroscopic conditions. The scaling is necessary because the subgrid trajectories will, in general, be generated using an imposed hydraulic gradient different from the macroscopic gradient resulting from the CPM flow simulation.

Format 2 is described as follows.

```
retdatadir
filename      ! filename of data file containing trajectory
                data. first character of filename needs to
                be nonnumeric
```

The **retdatadir** parameter provides the file system path to the generic lookup-table data required for sampling the retention time distributions. The parameter **filename** is the name of the data file that contains the trajectory data. The required format of the file named in *trajectories.dat* is the CONNECTFLOW modified verbose format for pathline calculations. That format allows for a single comment line. The format following the comment line is described as follows.

```
ntraj   REAL   TIMESTEP
  PART  TRAJ_NUM   REALISATION_NO   START TIME  flag
  xloc yloc zloc    XN XN XN   UX UY UZ   TOT_TIME
TOT_LENGTH TRAJ_BETA
   STEPS  num_seg
    CUM_TIME  xe ye ze  HYD_AP TRANS_AP tau beta cpmgradh
typeID
     : !repeat until the end of the trajectory is reached
     : !repeat until all of the trajectories have been
     entered
```

Several of the parameters in the input file are not employed by MARFA, but have been retained for consistency between MARFA and CONNECTFLOW. These unused data fields are labeled in uppercase lettering in the previous description; the unused data fields are expected as placeholders. The parameter **ntraj** is the number of trajectories. The parameter **flag** indicates whether the trajectory reaches a transport boundary; **num_seg** is the number of segments on the trajectory; and **xloc**, **yloc**, and **zloc** are the x-, y-, and z-positions for the trajectory head. The data read for each segment include the location of the end of the segment (**xe,ye,ze**); a five-character identifier for the rock type (**typeID**); the advective travel time (**tau**); the magnitude of the hydraulic gradient (**cpmgradh**), and the F-quotient (**beta**).

Three optional sections exist for the file *trajectories.dat,* independent of the format: "FLOW CHANGES", "VALID TRAJECTORIES" and "CHANNELING FACTOR". These three sections are key-phrase driven and may come in any order and are not required to be present. The optional keywords and associated input format are as follows.

```
CHANNELING FACTOR chanfac
FLOW CHANGES nchanges
 time_of_change  vscale
 : !above block repeated for each flow change to be
analyzed
VALID TRAJECTORIES
Trj_num terminal_segment
: !above block repeated for each trajectory to be analyzed
END
```

The optional key phrase "CHANNELING FACTOR" allows the user to specify a global factor **chanfac** that multiplies the segment beta values read from a PTV file (Format 2). This keyword has no effect in Format 1. The channeling factor also has no effect for rock types that are defined as having stochastic pathway variability (see Section A.3)

The key phrase "FLOW CHANGES" initiates an optional section that allows the user to specify velocity changes. The parameter **nchanges** is user defined, establishing the number of flow changes to be evaluated. The parameter **time_of_change** sets the time of a flow change (defining end of a flow period). The parameter **vscale** established the velocity change during the flow period. If the optional section "FLOW CHANGES" is not present or if **nchanges** is equal to zero, no flow changes are evaluated. Note under all conditions the number of retention parameter sets defined for each hydrofacies in *rocktypes.dat* must match the number of flow periods (**nchanges** + 1).

The key phrase "VALID TRAJECTORIES" initiates a section that allows the user to specify a subset of trajectories for particle transport analysis. The parameter **trj_num** identifies the index in the trajectory list of the trajectory to be evaluated. The parameter **terminal_segment** allows the setting of the monitoring location for each selected trajectory to a user-defined trajectory segment number. If the trajectory has fewer segments than the user-specified **terminal_segment** or if the user enters 0, the final segment of the trajectory will be used as the monitoring location. If the optional section "VALID TRAJECTORIES" is not present, all trajectories are evaluated, and the final segment of each trajectory is used as the monitoring location.

A word of caution regarding the specification of the terminal segment: if the trajectory input contains a segment with zero value for τ, the internal indexing of segments in MARFA will be different from the input. If MARFA encounters a zero-τ segment, then that segment is ignored, and indexing for subsequent segments in the trajectory is adjusted.

An example *trajectories.dat* file is shown in Figure A-5. This example uses Format 1 and has one trajectory with eight segments. The trajectory starts at the location (0,0,-6) and terminates at (11,0,250) after passing through the three rock types defined in Figure A-5. Note that the trajectory begins in an explicit DFN region, enters a repository tunnel, and then returns to an explicit DFN region before entering a CPM region. In the section FLOW CHANGES, one flow change has been specified, which occurs at 10,000 yrs with a relative change in velocity of 1.5. In the section VALID TRAJECTORIES, trajectory one has been included for analysis with breakthrough monitoring at segment seven.

```
/home/spainter/MARFA323/data/
1
CAN1  0.   0.    -6.0
       1   0.    -5.0       DFN1    3.0 2.0d4 0
      11   0.    0.         TUN1    100.  100.   0.0
      11.  0.    0.57073    DFN1    1.1105    48031.    0
      11.  0.    2.72963    DFN1    0.00095129    36.061    0
      11.    0.    3.62113    DFN1    3.0657    19465.    0
      11.    0.    4.97553    DFN1    3.8401    519210.    0
      11.    0.    10.0       DFN1    3.7925    88610.    0
      11.    0.    250.       CPM1    100.  1.0d6  0.001
 END
FLOW CHANGES  1
10000.0  1.5
VALID TRAJECTORIES
   1  7
END
```

An example *trajectories.dat* file for format 2 is shown in Figure A-6. The file containing the trajectory data is *nro.ptv*. In the section “FLOW CHANGES”, no flow changes have been defined. In the section “VALID TRAJECTORIES”, three trajectories will be used in the particle transport calculations. Trajectories 1, 15, and 31 terminate at segments 100, 150, and 120, respectively.

A.6 The Subgrid Trajectories

For each stochastic rock type, the user must supply sets of subgrid trajectories, one set for each direction. As discussed in Section 2, the subgrid trajectories are used to recover the effects of subgrid scale variations in velocity, which were lost in the CPM representation of flow. Specifically, the subgrid trajectories provide the raw data for the Painter and Cvetkovic [2005] upscaling/downscaling algorithm. The full path name for the directory containing the subgrid trajectories is read from the *rocktypes.dat* file. Contained

```
/home/spainter/MARFA323data/
nro.ptv
FLOW CHANGES 0
CHANNELING FACTOR 0.1
VALID TRAJECTORIES
   1     100
   15    150
    31   120
END
external CONNECTFLOW file (Format 2).
```

Figure A-6. *Example trajectories file with trajectories contained in an in that directory are several subdirectories, one for each stochastic rock type. Each subdirectory must be named as the corresponding rock type identifier (**typeID**). Each subdirectory must contain several files, one for each direction in the direction set. The file name for i-th direction must be formed by appending _i to the **typeID**. Thus, the subgrid trajectory file for direction 2 of a rock type named CPM4 must be named CPM4_2.*

The format for each subgrid trajectory file is

```
!header information
:
gradh dip strike vcpm ! imposed gradient and upscaled
transport velocity
nsgt                  ! number of subgrid trajectories in
this set
tau beta length       ! segment data
:                     ! repeat above line for each segment
END                   ! end-of-trajectory flag
:                     ! above block repeated for each
                      trajectory
```

Here **dip** and **strike** are the dip and strike angles (in degrees) characterizing the direction of the imposed hydraulic gradient used in constructing the given set of trajectories; **gradh** is the magnitude of the imposed gradient. The magnitude of the upscaled transport velocity is denoted **vcpm**. The parameter **nsgt** is the number of trajectories in the set. For each segment the parameters **tau**, **beta**, and **length** are read. These represent the advective travel time, F-quotient, and length for each segment, respectively. The END keyword denotes the end of a trajectory.

MARFA pre-processes the subgrid trajectory information to a form convenient for use by the downscaling algorithm. Because the pre-processing step may take several minutes for large sets of subgrid trajectories, MARFA saves the results of the pre-processing in intermediate files. Previously processed subgrid data may then be used in subsequent runs without re-running the preprocessor.

The existence of pre-processor output in the subgrid trajectory directories controls whether the pre-processor runs. If the subgrid trajectories directory contains pre-processed data files and information extracted from the headers of these files is consistent with certain input from the *rocktypes.dat* file, then the pre-processor is skipped. Otherwise, it is called by MARFA to process the raw subgrid trajectories.

The naming convention for the processed subgrid data is identical to that of the subgrid trajectories, but without the rock type name. For example, the file *_1* in the *CPM1* directory is the processed subgrid data for direction 1 of rock type CPM1. The file *CPM1_1* in the same directory would contain the raw subgrid trajectories.

A.7 Postprocessor Control

The postprocessor calculates breakthrough curves based on the particle arrival times. The user may specify which trajectories are to be included in the breakthrough curve calculation. In addition, the times at which breakthrough values are needed may also be specified. The postprocessor options are read from the *postprocessor.dat* file. If the file *postprocessor.dat* is not present, the postprocessor will include all trajectories and calculate discharge at 200 times, spread uniformly between the maximum and minimum arrival times for each nuclide. In this mode of operation, postprocessor operation is fully automated and requires no user input.

The user may opt to include all trajectories or restrict the breakthrough curve to a subset of the trajectories. If a subset is required, the user must specify which trajectories are of interest by defining one or more batches of trajectories, where a batch is defined by a continuous range of indices in the trajectory list. A batch is specified by entering the lower and upper indices (inclusive) in the trajectory list.

Three options are provided for determining discharge (breakthrough) calculation times. The times may be determined automatically based on a uniform or logarithmic spacing, or specified manually.

The format for the *postprocessor.dat* file is as follows.

```
gamma t0   ! sensitivity parameter and time of first release
trajset    ! Enter A for all or P for a subset
nbatches   ! number of batches if Partial list selected
i1  i2     ! upper and lower indices defining a subset
:          ! repeat for total of nbatches

disttype    ! distribution type uniform, logarithmic, manual
(U/L/M)
nbrktimes  ! number of times at which breakthrough values
are needed
```

```
time1    ! if manual is selected first time
:        ! repeat for a total of nbrktimes
```

Gamma is a sensitivity parameter used by the postprocessor when reconstructing the instantaneous breakthrough curves. The default value for gamma is 0.2, which under most conditions results in an accurate and robust breakthrough curve. Under conditions that may result in prompt or rapid changes in breakthrough, however, larger values of gamma (e.g., 0.3 or 0.4) more accurately track the rapid transitions. Increasing the sensitivity will result in more statistical noise throughout the entire breakthrough curve.

The optional parameter **t0** is the first time at which releases into the far field are possible. This parameter allows the user to redefine the time "origin" and enforce a zero-release condition at that time. In a pinhole release case, for example, t0 would be set to the time that the hole appears.

The **trajset** parameter indicates whether the breakthrough will be based on all trajectories or a subset. Allowed values are A for all trajectories or P for a partial reconstruction based on a subset of the trajectories. If a partial reconstruction is specified, **nbatches** is the number of batches; **i1** and **i2** are the lower and upper indices for the batch.

The **disttype** parameter controls the times at which breakthrough curves are calculated. Allowed values are U for uniformly spaced between minimum and maximum arrival time, L for logarithmically spaced, and M for manually specified. The parameter **nrbktimes** is the number of times at which discharge is required. If times are manually specified, a list of length of **nbrktimes** must be given, one per line.

APPENDIX B: MARFA 3.3.1 INPUT

MARFA 3.3.1 requires several input data files. Files that define the radionuclide decay chains, the rock retention properties, and the postprocessor control are *nuclides.dat*, *rocktypes.dat,* and *postprocessor.dat*, respectively. These files are the same as the corresponding files for MARFA 3.2.3. The *source.dat* file contains radionuclide source histories and is only slightly different from the corresponding Version 3.2.3 input. In addition to those files, MARFA 3.3.1 requires files that describe the node networks, the time of flow changes, and the number of visits a particle can make to a node before it is terminated. A translation table for each flow change is required if nodes are numbered differently in the two time periods. If nodes are numbered the same in every flow period, the node translation tables may be omitted.

B.1 The flowdata.dat File

Information about the flow periods is contained in the *flowdata.dat* file. This file is required and must have the name *flowdata.dat*. The format of the *flowdata.dat* is

```
datadirectory
VISITS ALLOWED maxvisits
FLOW PERIODS nfp
Time iufp              ! one time needed for each flow
                     period
NODE DATA FILES nufp
nodedatafile           ! one file needed for each flow
                     period
TRANSLATION DATA FILES
translationtablefile   ! one table needed for each flow
change
```

The character string **datadirectory** is the path to the directory containing the lookup tables, as in Version 3.2.3. This character string is required and must be on the first line. The remainder of the **flowdata.dat** file is key-phrase driven. The valid key phrases are “VISITS ALLOWED,” “FLOW CHANGES,” “NODE DATA FILES,” “TRANSLATION DATA FILES,” and “CHANNELING FACTOR.” The key phrases may come in any order after the first line. Each key phrase is followed immediately by corresponding input.

The integer **maxvisits** following the optional key phrase “VISITS ALLOWED” is the number of times a particle may visit a node. If a particle visits a node more than **maxvisits** times, the particle is terminated as described in Section 2.2. If the “VISITS ALLOWED” key phrase is not present in the file, **maxvisits** defaults to 1.

The integer **nufc** immediately following the required key phrase “NODE DATA FILES” is the number of unique flow periods. Following the key phrase, a total of **nufc** values of the character string **nodedatafiles** are read. These strings

specify the locations of the node network data files in the MARFA 3.3.1 format (see Section 3.2). A total of **nufc** files are required, one for each flow period.

The integer **nfp** following "FLOW PERIODS" is the number of flow periods during the simulation. After this keyword is read, **nfp** values of the **time** and **iufp** fields are then read, one value of each per line. Each value of the **time** field is the time in years at the end of a flow period. The first period is presumed to start at time 0. The **iufp** are indices into the node data file. For example, a value of 2 for **iufp** means the flow period in question is associated to the second node data file. Note that each **iufp** must be less than or equal to **nufp**.

The **translationtablefile** character string specifies the location in the local file system of the data file that maps corresponding nodes between different **nodedatafiles**. If the "TRANSLATION DATA FILES" keyword is provided, a translation table is required for each flow change. If the "TRANSLATION DATA FILES" key phrase is not present, the identity mapping is used for each flow change.

The optional key phrase "CHANNELING FACTOR" allows the user to specify a global factor **chanfac** that multiplies the segment beta values in the **nodedatafiles**.

B.2 The Node Network Data Files

Information about the groundwater velocity field in each flow period is provided to MARFA 3.3.1 in the form of a directed graph, which provides node-to-node connectivity. The format of a node network data file is

```
num_nodes
    node_num   xloc  yloc  zloc     !beginning of
individual node data
    num_neigh
    neigh_i        …   !repeated a total of num_nodes times
    prob_i         …
    tau_i          …
    length_i       …
    beta_i         …
    rocktypeID_i   …
    : ! Repeat node data structure for each node
```

The field **num_nodes** is the total number of nodes contained in the data file. The integer field **node_num** and the real fields ***x***, ***y***, ***z*** are the node number and its spatial coordinates, respectively. The integer **num_neigh** is the number of nodes immediately downstream of the current node. The node numbers for downstream nodes are given in the **neigh** fields. A total of **num_nodes** must be given, all on one line. The **prob** fields are the probabilities of advancing to each downstream node. The probabilities are input as a cumulative probability for neighbors 1 through **num_nodes**. The pathway to each downstream node is defined by a transport time, length, retention parameter, and rock type, which

are **tau**, **length**, **beta,** and **rocktypeID,** respectively. The definitions of these parameters are the same as in Version 3.2.3.

A helper application named *extractnodedata* is included with MARFA 3.3.1. The *extractnodedata* application prompts for a NAPSAC extended ASCII format (NEF) file describing the fracture system and a CONNECTFLOW binary pathline library (PTH) file. The application writes a file named *nodedata.dat* in the correct format for MARFA 3.3.1.

B.3 The Node Remapping Files

Because of the possibility of different node numberings among multiple data sets, MARFA 3.3.1 uses a translation table to map nodes between data sets. This table provides a simple node-to-node mapping between node network data sets before and after a flow change.
The file format is

```
node1a          node1b
node2a          node2b
   :               :
! repeat a total of num_nodes times
```

Node$_i$a and **Node$_i$b** are node numbers, with **Node$_i$a** the node number in the earlier flow epoch and **Node$_i$b** the node number of the later flow epoch to which **Node$_i$a** will be mapped. If a node does not have a corresponding node in the later flow epoch, the node is mapped to a fictitious zeroth node (i.e., **Node$_i$b** = 0).

B.4 The *source.dat* File

The radionuclide source is specified in the *source.dat* file. The source is to be specified as a source rate in mol/yr or Bq/yr. The format is nearly identical to Version 3.2.3.

```
npart
units           ! mol/yr or Bq/yr
samplemethod    ! keyword controlling source sampling
nsources        ! number of sources
sourceID        ! beginning of source block
nnodes          ! number of nodes associated with this
                source
nodenum         ! node number for node associated with
                source
:               ! repeat for a total of nnodes lines
ndpts
time RN1 RN2 RN3 …          ! one for each nuclide
 :         ! repeat above line for a total of ndpts lines
:! repeat above source block for a total of nsources blocks
```

The parameter **npart** is the total number of particles to be used. The **units** parameter is used to specify the source units. The options are “mol/yr” or “Bq/yr.” The optional keyword **samplemethod** controls the sampling of the source. The default method is to sample the source history with uniform statistical weights for each particle of a given radionuclide. If the optional **samplemethod** keyword is present and specified as “UNIFORM IN TIME,” sampling of the source will be uniform in time with appropriate statistical weighting applied to each particle, as described in Section 3.3 of this document. The parameter **nsources** is the number of source locations (e.g., failed canisters). For each source, the **sourceID** is a 10-character identifier for that source. The parameter **nnodes** is the number of nodes associated with the specified source. The node numbers **nodenum** associated with the source are then read, one per line. The number of time points in a source history is **ndpts**. For each value of **time**, **nnuc** values of the source release rate are read, where **nnuc** is the number of nuclides.

APPENDIX C: MARFA OUTPUT

MARFA does not have a graphical user interface. Error and warning messages are written to the screen. MARFA also reports progress in initializing the calculation. Once the main Monte Carlo calculation starts, no output is provided until after all particles are processed.

Raw results in the form of particle arrival times are written to the file *results.dat* in the run directory. If the file does not exist, it will be created. If it does exist, it will be overwritten. The *results.dat* file starts with several lines of header information. Following the header, the output format is

```
nnuc                              ! number of nuclides
nuc_name  lambda                  ! nuclide name, decay constant
:                                 ! repeat a total of nnuc times
atime inuc mass sourceID trajID  ! data for each particle
:                          ! repeat for each particle arriving
```

Here **nnuc** is the number of nuclides modeled; **nuc_name** and **lambda** are the name and decay constant for each nuclide. One line of data is written for each particle that makes it to a trajectory terminus; **atime** is the arrival time and **mass** is the mass in moles of the particle. The integer **inuc** is an index into the nuclide list, and thus specifies the state of the particle upon arrival. The character string **sourceID** uniquely identifies the source location for the particle from the lists in Sections 5.2 and 5.4 and the **trajID** uniquely identifies the trajectory for the particle.

A post-processor calculates instantaneous and cumulative breakthrough curves from the arrival times. The file *Breakthrough_Mo.rlt* contains instantaneous mass discharge in units of mol/yr and cumulative mass discharge in units of mol. The file contains a global header section followed by several results sections, one for each nuclide type. The format following the global header section is

```
!ID =1    Name= nname1 Number of particles = nscore1
!Time Cumulative Breakthrough Instantaneous Breakthrough
!    (Moles) (Moles/year)
!
time1                cbrkval1            ibrkval1
:          ! repeat above line for each estimation time
:          ! repeat above block for each nuclide modeled
```

Where **nname1** is the name of nuclide 1, and **nscore1** is the number of particle "scoring" (arriving at a pathway endpoint) as nuclide 1. The cumulative and instantaneous breakthrough curves are printed following that four line header, with **cbrkval** and **ibrkval** representing cumulative and instantaneous breakthrough at a given time (**time**).

The file *Breakthrough_Bq.rlt* is identical to *Breakthrough_Mo.rlt* except that results are in units of Becquerel per year.